ISSN 2313-7304

ПРОЦЕССЫ УПРАВЛЕНИЯ И УСТОЙЧИВОСТЬ

ТОМ 12 (28)

ТРУДЫ LVI МЕЖДУНАРОДНОЙ НАУЧНОЙ КОНФЕРЕНЦИИ АСПИРАНТОВ И СТУДЕНТОВ

Санкт-Петербург
7 – 11 апреля 2025 года

ISSN 2313-7304

CONTROL PROCESSES AND STABILITY

Vol. 12 (28)

PROCEEDINGS OF THE LVI INTERNATIONAL CONFERENCE

St. Petersburg
April 7 – 11, 2025

PUBLISHING HOUSE
FEDOROVA G.V.
2025

ББК 22.1
П84

Редакционная коллегия:

д-р физ.-мат. наук, проф. Л. А. Петросян (отв. редактор) (СПбГУ),
д-р физ.-мат. наук, проф. Н. В. Смирнов (зам. отв. редактора) (СПбГУ)
д-р физ.-мат. наук, проф. О. И. Дривотин (СПбГУ)
д-р физ.-мат. наук, проф. А. П. Жабко (СПбГУ)
д-р физ.-мат. наук, проф. Е. Д. Котина (СПбГУ)
д-р физ.-мат. наук, проф. А. Ю. Крылатов (СПбГУ)

Процессы управления и устойчивость. Том 12(28). № 1 / науч. П84 редакторы тома Н. В. Смирнов, А. С. Еремин, Т. Е. Смирнова. СПб.: Издательский Дом Федоровой Г.В., 2025. 632 с.
ISSN 2313-7304

«Процессы управления и устойчивость» – ежегодное научное издание. Издается на факультете прикладной математики – процессов управления СПбГУ с 1998 года. Том журнала традиционно формируется по результатам работы ежегодной Международной научной конференции аспирантов и студентов «Процессы управления и устойчивость» и включает статьи студентов, аспирантов, сотрудников СПбГУ и других высших учебных заведений, в том числе зарубежных, по математической теории процессов управления, математическим методам в механике и физике, математическому моделированию в медико-биологических системах, информационным и компьютерным технологиям, теории управления социально-экономическими системами.

В Том 12(28) включены статьи участников 56-й Международной научной конференции аспирантов и студентов «Процессы управления и устойчивость» (7 – 11 апреля 2025 года), прошедшие рецензирование и рекомендованные к публикации. Журнал предназначен для студентов старших курсов физико-математических факультетов университетов, аспирантов и научных работников.

ББК 22.1

Журнал издается при финансовой поддержке
Факультета прикладной математики – процессов управления
Санкт-Петербургского государственного университета

ISSN 2313-7304

© Издательский Дом Федоровой Г. В., 2025
© Смирнов Н. В., 2025

Содержание

Пленарная сессия **11**

Приглашенные лекторы 11

Сидорин А. О. Процессы управления в фундаментальной науке 11

Аббасов М. Э. Построение оптимальной траектории на местности 13

Ващенков К. В., Пономарев Ю. Ю. Платформа компании XSQUARE-LCDP – российская Postgres-ориентированная платформа для low code систем – путь в будущее для начинающих разработчиков. Краткое введение 27

1. Математическая теория процессов управления **37**

Евтина Д. С., Жабко А. П. Оценка перерегулирования уравнений нейтрального типа с несоизмеримыми запаздываниями 37

Живулин Е. А., Гришкин В. М. Построение маршрута движения судна с помощью методов минимизации с целью экономного расхода топлива 42

Каминцева В. С. Построение оптимального регулятора для систем с возмущениями 47

Костеров М. А. Вычисление коэффициентов параболических аппроксимационно-оценочных критериев при помощи QR-разложения 55

Семенова А. Ю. Многокритериальная оптимизация параметров управления динамикой пучка заряженных частиц в линейном ускорителе 62

Соколов В. М. Функционал Ляпунова – Красовского с заданной производной для волнового уравнения 71

Эстрада Хименес О. Алгоритм определения последовательности движений, необходимых для изменения положения штабелируемых контейнеров 78

2. Математические методы в механике и физике 85

Вавренюк В. В. Нечеткий регулятор в задаче стабилизации солнечного паруса 85

Викторов Б. В. Алгоритм обхода группой роботов набора точек 95

Вяльцева В. В. Оценка наведенной анизотропии эффективных свойств материалов с трещинами 100

Гумеров И. И., Холодова С. Е. Разработка методов математического моделирования динамических процессов во вращающейся электропроводящей жидкой среде 105

Кикенов А. А. Моделирование управляемого движения конвертоплана-трикоптера 112

Князев Н. А., Левинский Д. В. Учет изменений магнитных полей диполей при моделировании поперечного движения частиц в синхротроне 118

Муллаянов Т. О. Влияние формы наноразмерного рельефа на распределение напряжений вблизи свободной поверхности твердого тела 124

Ручьев И. А., Шиманчук Д. В. Методы машинного обучения для решения обратной задачи кинематики робота-манипулятора с избыточными числом степеней свободы 130

Сулейманов В. Н. Моделирование формы солнечного паруса под действием давления солнечного света в окрестности точки либрации L1 139

Якимова И. О., Шувалов Г. М. Моделирование распространения трещины под действием напряжений и коррозии 147

Li Y. Design and implementation of Multi-UAV cooperative reconnaissance and strike mission planning system 152

3. Математические модели медико-биологических систем 157

Аникушин В. А., Редько А. Ю. Прогнозирование динамики эпидемии COVID-19 на основе процентных приростов в странах Европы и Азии 157

Васильев Е. В. Тепловые карты, сопряжённые с иерархической структурой медицинских наблюдений 165

Давыденко Б. Н. Текстурный подход к анализу микрофотографий клеточных культур 170

Кириченко Е. Ю., Громов Р. М., Блеканов И. С., Назаров В. Д., Денисов Д. Г., Лапин С. В. Методы диагностики анемии в лабораторных исследованиях на основе машинного обучения . 174

Козин А. А. Классификация новообразований молочной железы по BI-RADS с помощью нейронных сетей 179

Кривошеин А. Е., Хоменко И. Е. Многоклассовый и двухэтапный подход к классификации и обработке медицинских данных 184

Моргач Д. Ю. Фильтрация электромиографических сигналов с целью классификации действий 189

Топпер А. М. Разработка автоматизированной системы обработки амбулаторных карт пациентов 194

Шайхисламова А. Р. Математическое моделирование динамики смертности в Республике Молдова: вероятностно-статистический анализ 199

Шарашева Н. И. Прогнозирование распространенности онкологических заболеваний 205

Katguia F. H. AI-explainable medical image comparison 211

Wang B., Hu R., Xia Y. Rumor spread model and impact assessment in public health mergencies 218

4. Информационные и компьютерные технологии 223

Авраменко П. А. Приближенные раскраски графов в телекоммуникационных сетях 223

Аржа А. Н. Применение автокодировщиков для поиска аномальных временных рядов в наборе данных с умных счетчиков 229

Аристакесян Т. А., Блеканов И. С. Методы декомпозиции тем пользовательской дискуссии на основе динамического тематического моделирования 236

Бабкина А. В., Вахитов Б. Д., Девришев Т. Э., Мишин И. В., Петренко А. А. Детектирование признаков туберкулеза на рентген-флюорографиях легких 243

Бабкина А. В., Конев Д. В. Сегментация почек на ультразвуковых изображениях 251

Баграмов Н. И. Автоматическая генерация аннотаций научных публикаций на основе методов обработки естественного языка 259

Васильев В. А. Квантизация весов в ключевых компонентах трансформерных архитектур 265

Вашукова А. М. Применение нейронных обыкновенных дифференциальных уравнений для моделирования динамических систем 270

Гаврилов Н. О., Корхов В. В., Пен Е. А., Токарев А. Э. Разработка алгоритма построения графа знаний на основе патентных данных 275

Голофастов Л. Д., Миронов А. В. Реализация и анализ мощности критериев согласия с экспоненциальным распределением 282

Голощапова А. М. Применение методов оптимизации гиперпараметров для повышения точности моделей прогнозирования кликов рекламы 290

Жариков Д. С., Блеканов И. С. Методы распределенной загрузки медицинских данных в формате КТ-снимков 295

Зубрицкий И. В. Эффективный по занимаемой площади метод реализации аппаратной поддержки криптографической защиты информации 301

Ибатуллин Е. В., Ламзин Н. А. Архитектура Data Mesh для среды больших данных 309

Измайлов Н. О. Обучение бинарной нейронной сети с использованием ее непрерывного аналога 313

Круглов Н. А. Программный комплекс для решения задач оптимизации перевозок 319

Кузнецова Д. С. Исследование методов инициализации алгоритма роя частиц (PSO) для решения задач глобальной оптимизации 325

Кукло Р. А. Исследование методов сбора, обработки и сжатия данных с помощью граничных вычислений в AIoT 333

Лаврухин М. Ю. Применение генетического алгоритма для оптимизации трёхмерных траекторий на рельефе 340

Миронов А. В. Имплементация и анализ мощности критериев согласия с распределением Вейбулла 346

Мошкин Л. С. Использование аппроксимационно-оценочных критериев для модификации алгоритма HDBSCAN 352

Мурадян Д. С., Алимов П. Г. Реализация и внедрение архитектуры Gated Recurrent Unit в пакет нейросетевой аппроксимации дифференциальных уравнений DEGANN 360

Петрова А. М. Методы распознавания различных видов продуктов питания на изображениях 366

Пономарев Ю. Ю. Вопросы точных оценок характеристик деревьев при моделировании сверхбольших графов технологических и естественных сетей с заданными статистиками вершин, ребер и прохождения сигналов 372

Попов А. П. Методы построения интеллектуальных ассистентов на основе LLM для обработки открытых медицинских данных 379

Похабов И. И., Гориховский В. И. Графовые критерии проверки гипотез согласия 385

Саперов Д. П. Оценка влияния мировых событий на рынок криптовалют 391

Смирнов Д. О., Блеканов И. С. Методы адаптации больших языковых моделей на примере домена LegalTech 397

Снапков М. Ю., Шмаков А. А. Применение данных социальных сетей в задачах анализа финансовых инструментов 405

Соловьев Д. Н. Система домашней автоматизации с использованием децентрализованного подхода в хранении и обработке данных 411

Чайкин Г. А. Создание синтетических данных пользовательской активности на основе вопросно-ответных текстовых данных 417

Шабалов Н. А., Бойко А. В., Малков А. В. Применение ML алгоритмов для модели тарификации ОСАГО 422

Шайдуров В. Д., Корхов В. В., Половков В. В., Пен Е. А., Карельский К. В. Разработка системы анализа и картирования твердых полезных ископаемых морского дна с использованием искусственного интеллекта 432

Юрьевцева П. Н. Исследование чувствительности LLM к психоэмоциональным параметрам пользователей на основе профилирования синтетических персон 441

Fan J., Liu J., Davydenko A. A. Applying MegaAgent framework for oil and gas extraction optimization 447

Fan X., Blekanov I. S., Gubar E. A. Evolutionary approach based on BERTopic and NetLogo to analyse user discussions in social media 455

Feng Y. Obtaining estimates for the rate of convergence of the gradient descent via machine learning 461

Gao T. Structural and power-law degree distribution analysis of public transport networks in Chinese and Russian cities based on openstreetmap data and complex network theory 467

Gao Y., Xu H. Reliability of communication network 476

Li J., Blekanov I. S. Text encoding approaches for topic modeling in the social media platform 485

Liu J., Fan J., Davydenko A. A. Swin-RepGFPN synergized YOLO: A dangerous driving behavior detection in near real-time . 491

Wang Q., Gao Z., Zhang L. Research on tomato maturity detection model based on YOLOV11 in complex lighting scenarios 499

Xu H., Petrosian O. L. Printed circuit board defect detection based on improved RT-DETR 504

Zhang L., Wu H., Petrosyan O. L. Online adaptive forecasting for web traffic with sudden drift 510

5. Управление социально-экономическими системами..517

Ван Д., Смирнов Н. В., Го Ч. Модель оптимального управления логистикой торговой фирмы 517

Золотухина П. Ю. Сравнение методов CPM и PERT планирования сроков выполнения проекта на примере организации выставки фотографий 524

Ибраева А. С. Оценка эффективности модифицированного NSGA-II в задаче о назначениях: сравнение с точным методом и классическим NSGA-II 528

Карабаев Е. Е. Анализ выпуска продуктов птицеводства с помощью динамических систем 536

Кремко В. В. Анализ и оценка эффективности методов машинного обучения в прогнозировании цен на продукты питания 543

Мова И. А. Анализ потребительских и ипотечных кредитов по регионам России: пространственно-эконометрический подход 550

Новгородцев В. А. О задаче управления портфелем в случае модели Бейтса 557

Субботина Е. А. Оптимизация затрат на рекламу в социальных сетях с использованием моделей влияния 562

Фролова Е. А. Модель энергетического микса в Российской Федерации 567

Cao Z. Dynamic networks with complete information and incomplete information under cross-partitioning 572

Du L. The stability of competing information flow in epidemic models 577

Kong Q., Petrosyan L. A. Dynamic network game with changing communication structure 582

Pu Y., Vostrenkova E. M. A framework integrating environmental feedback and agent opinions. Evolutionary game theory and reinforcement learning 587

Qu R., Huang R. Integrating Raft algorithm to blockchain consensus mechanism 592

Sun S., Petrosyan L. A. Dynamic hierarchical games with changing communication structure and pairwise interactions 597

Xiong X. A history-driven dynamic network game model for smart home systems: Comparative analysis with Q-learning 602

Xu Y. Cooperative network game with partner sets 606

Yu X. An evolutionary games model with time delays 611

Zheng Z. Impact of interaction time on overprotective behavior during epidemics: an evolutionary model 616

Contents **622**

Именной указатель **629**

Пленарная сессия



Сидорин Анатолий Олегович

кандидат физико-математических наук,
лауреат Премии РАН имени академика В. И. Векслера,
кафедра информационных и ядерных технологий,
факультет прикладной математики — процессов управления,
Санкт-Петербургский государственный университет



Процессы управления в фундаментальной науке

Аннотация. Большинство исследований фундаментального характера требует создания крупных комплексов экспериментальной аппаратуры – так называемых установок класса мегасайенс. Независимо от области исследования таким комплексам необходимы программные средства для моделирования физических процессов, распределенные системы хранения и обработки больших объемов данных, современные системы управления с элементами искусственного интеллекта. Возможные пути решения этих задач иллюстрируются на примере комплекса NICA, создаваемого в Объединенном институте ядерных исследований.

Sidorin Anatoliy Olegovich

PhD, Associate Professor,
the Veksler Prize of the Russian Academy of Sciences,
Department of Information and Nuclear Technologies,
Faculty of Applied Mathematics and Control Processes,
Saint Petersburg State University

Control processes in fundamental science

Abstract. Usually fundamental researches require creation of large complexes of the experimental equipment – so called mega-science facilities. Independently on the field of investigations, each such facility requires computer tools for simulation of physical processes, systems for distributed storage and processing of big amount of data, modern control systems with elements of artificial intelligence. Possible ways for solution of these tasks are illustrated on example of the NICA project, which is under realization in the Joint institute for nuclear research.

Аббасов Меджид Эльхан оглы

доктор физико-математических наук, профессор,
кафедра математической теории моделирования систем управления,
факультет прикладной математики — процессов управления,
Санкт-Петербургский государственный университет



Построение оптимальной траектории на местности

Аннотация. В докладе рассматривается задача определения оптимальной по стоимости строительства траектории дороги, соединяющей две фиксированные точки на местности. На практике такие проблемы часто решают с помощью различных эвристических процедур. Для получения теоретически обоснованного результата можно математически строго формализовать задачу, выведя интегральный функционал стоимости. Такой подход приводит к проблеме вариационного исчисления с закрепленными концами. Необходимое условие минимума такого функционала выражается в виде интегро-дифференциального уравнения. Будут обсуждаться вопросы существования и единственности решения этого уравнения, а также различные методы получения решения.

Abbasov Majid Elkhan ogly

D.Sc., Professor,
Department of Mathematical Theory of Modeling of Control Systems,
Faculty of Applied Mathematics and Control Processes,
Saint Petersburg State University

Searching for optimal trajectory on the terrain

Abstract. The report considers the problem of determining the optimal cost of constructing a road trajectory connecting two fixed points on the terrain. In practice, such problems are often solved using various heuristic methods. In order to obtain a theoretically sound result, the problem can be mathematically formalised by deriving an integral cost functional. This approach leads to the problem of variational calculus with fixed ends. The necessary condition for the minimum of such a functional is expressed as an integro-differential equation. The existence and uniqueness of a solution to this equation are discussed, as well as various methods for obtaining a solution.

УДК 517.9

Аббасов М. Э.

Построение оптимальной траектории на местности

Введение. Задача построения оптимальной по стоимости строительства траектории, соединяющей две заданные точки на рельефе местности, возникает при проектировании транспортной инфраструктуры между двумя населенными пунктами, месторождением сырья и пунктом его переработки и т. д. Такие проблемы решаются с помощью геоинформационных систем (ГИС) – специализированного программного обеспечения, использующего графовые алгоритмы. В рамках этого подхода сначала строится сетка стоимости: исследуемая область разделяется на подобласти, каждой из которой присваивается стоимость ее пересечения. Далее к построенной сетке можно применить алгоритм полного перебора всевозможных траекторий, соединяющих начальную и конечную подобласти, и получить оптимальный маршрут. Такой метод работает лишь для сеток небольшой размерности. При попытке повысить точность полученного решения неизбежно приходится увеличивать плотность сети, повышая ее размерность. Это приводит к драматическому росту вычислительной сложности, так как количество арифметических операций, требуемых для реализации полного перебора имеет оценку $O(n^3)$, где n – размерность сетки. В результате при больших n задачу удается решить только с помощью некоторых эвристических подходов, таких например, как A*, D*. Однако алгоритмы этого класса приводят, вообще говоря, лишь к квазиоптимальному решению, поэтому они непригодны для получения решения с любой требуемой точностью. Таким образом, возникает противоречие – для получения более точного решения требуется повышать размерность сетки, однако для таких задач можно использовать лишь подходы, не способные выдавать точное решение.

В данной работе представлен новый подход к решению задачи нахождения оптимальной по стоимости траектории. За счет предложенной формализации проблемы удалось свести ее к задаче вариационного исчисления: при некоторых дополнительных предположениях описан функционал стоимости, которому должна доставлять минимум оптимальная траектория. Это дало возможность применить

Аббасов Меджид Эльхан оглы – профессор, Санкт-Петербургский государственный университет; e-mail: m.abbasov@spbu.ru, тел.: +7(812)363-65-37

к решению задачи весь богатый арсенал вариационного исчисления, а также численных методов, позволив быстро получать результат с любой требуемой точностью.

1. Постановка задачи и ее формализация. Пусть даны две заданные точки, которые нужно соединить траекторией с минимальной стоимостью строительства. Опишем вначале предположения, в которых будем работать [1].

1. Общая стоимость строительства складывается из стоимости доставки строительных материалов и стоимости строительных работ (заработная плата работникам, аренда техники и т. д.).

2. Доставка строительных материалов всегда осуществляется из начальной точки и производится по уже построенному участку пути.

3. Технология строительства одинакова в любой точке траектории.

4. Перепад высот на местности, где ведется строительство, незначительный.

Для формализации задачи на основе этих допущений, нужно ввести параметры, характеризующие стоимость доставки строительных материалов, а также самих строительных работ.

Обозначим α – стоимость доставки, приходящейся на единицу длины пути (от начальной точки), объема строительных материалов, необходимых для укладки единицы длины траектории. Благодаря третьему предположению, можно утверждать, что $\alpha = \text{const}$.

Каждая точка имеет свои условия строительства, обусловленные рельефом, ландшафтом и другими факторами. Поэтому вводится функция $\beta(x, y)$ – отплата труда работников, затраты на аренду техники и прочих, сопутствующих укладке работ за единицу длины пути.

Рассмотрим декартову систему координат с началом в точке O. Пусть $(l, 0)$ – координаты конечной точки A, а $y\colon \mathbb{R} \to \mathbb{R}$ произвольная непрерывно-дифференцируемая функция, удовлетворяющая граничным условиям

$$y(0) = 0, \quad y(l) = 0. \tag{1}$$

Любую такую кривую будем называть допустимой. При сформулированных предположениях функционал стоимости строительства дороги, определяемой функцией $y(x)$, имеет вид

$$J(y) = \int_0^l \alpha\sqrt{1+y'^2(x)} \int_0^x \sqrt{1+y'^2(\xi)}\, d\xi\, dx + \\ + \int_0^l \beta(x,y)\sqrt{1+y'^2(x)}\, dx, \tag{2}$$

где $\beta\colon \mathbb{R}^2 \to \mathbb{R}$ – заданная функция с непрерывными производными до второго порядка включительно.

Лемма. *Для произвольной функции $f(x) \in \mathbb{C}[0,l]$ справедливо равенство*

$$\int_0^l f(x) \int_0^x f(\xi)\, d\xi\, dx = \frac{1}{2}\left(\int_0^l f(x)\, dx\right)^2.$$

Используя лемму, перепишем функционал (2) в более удобной форме:

$$J(y) = \int_0^l \alpha\sqrt{1+y'^2(x)} \int_0^x \sqrt{1+y'^2(\xi)}\, d\xi\, dx + \\ + \int_0^l \beta(x,y)\sqrt{1+y'^2(x)}\, dx = \frac{\alpha}{2}\left(\int_0^l \sqrt{1+y'^2(x)}dx\right)^2 + \\ + \int_0^l \beta(x,y)\sqrt{1+y'^2(x)}\, dx.$$

Приравняв к нулю первую вариацию полученного функционала (см. [2, 3]), получим необходимое условие минимума [1].

Теорема 1. *Для того чтобы на допустимой кривой $y_*(x) \in \mathbb{C}^1[0,l]$ достигался минимум функционала стоимости J необходимо, чтобы*

$$\frac{y_*''(x)}{1+y_*'^2(x)}\left(\alpha\int_0^l \sqrt{1+y_*'^2(x)}dx + \beta(x,y_*(x))\right) + \\ + y_*'(x)\frac{\partial\beta(x,y_*(x))}{\partial x} - \frac{\partial\beta(x,y_*(x))}{\partial y} = 0. \tag{3}$$

2. Существование и единственность решения. Исследуем вопрос существования и единственности решения полученного интегро-дифференциального уравнения (3). Вначале напомним вспомогательные результаты [4, 5].

Определение 1. Функции из множества $U \subset \mathbb{C}^1[0,l]$ называют равномерно ограниченными, если существует $p > 0$ такое, что

$$|y(x)| + |y'(x)| \leqslant p$$

для любого $y(x) \in U$ и любого $x \in [0,l]$.

Определение 2. Функции из множества $U \subset \mathbb{C}^1[0,l]$ называют равностепенно непрерывными, если для любого $\varepsilon > 0$ существует $\delta > 0$ такое, что для любых x_1, x_2 из $[0,l]$, для которых $|x_1 - x_2| < \delta$, и для всех $y(x) \in U$ имеет место неравенство

$$|y(x_1) - y(x_2)| + |y'(x_1) - y'(x_2)| < \varepsilon.$$

Теорема 2 (Арцела)**.** *Пусть $U \subset \mathbb{C}^1[0,l]$ множество непрерывно-дифференцируемых функций. Для того, чтобы U было компактным необходимо и достаточно, чтобы функции из множества U были равномерно ограничены и равностепенно непрерывны.*

Определение 3. Непрерывный оператор A, определенный на множестве U линейного нормированного пространства E с областью значений, расположенной в U, называется вполне непрерывным, если всякое ограниченное подмножество U он отображает в компактное.

Теорема 3 (Принцип неподвижной точки Шаудера)**.** *Если вполне непрерывный оператор A отображает ограниченное замкнутое выпуклое множество S банахова пространства на свою часть, то существует неподвижная точка этого отображения, т. е. такая точка $x \in S$, что $Ax = x$.*

Отметим, что задача эквивалентна операторному уравнению $Ay(x) = y(x)$, где оператор A, отображающий $\mathbb{C}^1[0,l]$ в $\mathbb{C}^1[0,l]$, задается следующим образом:

$$Ay(x) = \int_0^x d\xi \int_0^\xi f(\eta, y(\eta), y'(\eta))d\eta - \frac{x}{l}\int_0^l d\xi \int_0^\xi f(\eta, y(\eta), y'(\eta))d\eta.$$

Используя такое представление задачи, в работе [6] с помощью теоремы 3 был доказан следующий результат.

Теорема 4. *Пусть* $M \geqslant \dfrac{1}{2l}$ *и существует* $c \leqslant \dfrac{\alpha}{32l^2M^2}$ *такое, что функция* $\beta(x,y)$ *в области своего определения удовлетворяет условию*

$$\max\{|\beta_x(x,y)|, |\beta_y(x,y)|\} \leqslant c.$$

Тогда в области

$$G = \{y \mid \max_{x\in[0,l]} |y(x)| = \|y\|_{\mathrm{C}[0,l]} \leq 2l^2M, \quad \|y'\|_{\mathrm{C}[0,l]} \leq 2lM\}$$

существует решение уравнения (3), удовлетворяющее граничным условиям (1).

В работе [6] доказана также единственность решения:

Теорема 5. *Пусть выполнены все условия теоремы 4, а частные производные* $\beta_{xy}(x,y)$, $\beta_{yy}(x,y)$ *ограничены. Тогда в области* G *решение задачи (3), удовлетворяющее граничным условиям (1), единственно.*

Исследуемое интегро-дифференциальное уравнение (3) не может быть решено аналитически, поэтому далее рассмотрим несколько подходов, позволяющих получить решение.

3. Метод полиномиальной аппроксимации. Опишем основную идею подхода (см. [1, 7]):

- на отрезке $[0, l]$ введем равномерную сетку, содержащую $n+1$ узел;
- имея значения вторых производных искомой функции в узлах сетки, можно построить интерполяционный полином для $y''(x)$ степени n;
- интегрируя полученный многочлен и используя значения функции в первом и последнем узлах сетки (концах отрезка $[0, l]$), получаем интерполяционные многочлены степени $n+1$ и $n+2$ для функций $y'(x)$ и $y(x)$ соответственно;
- применяя какую-либо квадратурную формулу для вычисления интеграла $\displaystyle\int_0^l \sqrt{1+y'(x)^2}\,dx$, сводим проблему к задаче решения системы нелинейных уравнений относительно значений вторых производных в узлах сетки.

Обозначим через $p = (p_1, \ldots, p_{n+1})^T$ вектор коэффициентов интерполяционного многочлена для $y''(x)$. Пусть $\overline{y}^{(2)} = \left(y_1^{(2)}, \ldots, y_{n+1}^{(2)}\right)^T$ – вектор, компоненты которого равны значениям вторых производных в узлах сетки, т. е.

$$y_i^{(2)} = y''(x_i), \quad x_i = (i-1)\frac{l}{n}, \quad i = 1, \ldots, n+1.$$

Рассмотрим матрицу Вандермонда

$$Z = \begin{pmatrix} 1 & x_1 & x_1^2 & \ldots & x_1^n \\ 1 & x_2 & x_2^2 & \ldots & x_2^n \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 1 & x_{n+1} & x_{n+1}^2 & \ldots & x_{n+1}^n \end{pmatrix}.$$

Имеем $\overline{y}^{(2)} = Zp$, откуда $p = Z^{-1}\overline{y}^{(2)}$. Значит

$$y''(x) = XZ^{-1}\overline{y}^{(2)},$$

где $X = (1, x, \ldots, x^n)$.

Интегрируя последнее равенство в пределах от x_1 до x_i, для каждого $i = 1, \ldots, n+1$ получим

$$\overline{y}^{(1)} = Iy'(x_1) + S\overline{y}^{(2)},$$

где $I = (1, \ldots, 1)^T$– n-мерный вектор, $\overline{y}^{(1)}$ – вектор, компоненты которого равны значению функции $y'(x)$ в узлах сетки, и $S = BZ^{-1}$ – матрица, в которой

$$B = \begin{pmatrix} 0 & 0 & 0 & \ldots & 0 \\ x_2 - x_1 & \frac{x_2^2 - x_1^2}{2} & \frac{x_2^3 - x_1^3}{3} & \ldots & \frac{x_2^{n+1} - x_1^{n+1}}{n+1} \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ x_{n+1} - x_1 & \frac{x_{n+1}^2 - x_1^2}{2} & \frac{x_{n+1}^3 - x_1^3}{3} & \ldots & \frac{x_{n+1}^{n+1} - x_1^{n+1}}{n+1} \end{pmatrix}.$$

Обозначая $\overline{y}$ вектор, компоненты которого равны значению функции $y(x)$ в узлах сетки, аналогично приходим к равенству

$$\overline{y} = Iy(x_1) + SIy'(x_1) + S^2\overline{y}^{(2)}.$$

Учитывая, что в рассматриваемой задаче известно значение $y_n = y(x_{n+1})$ искомой функции $y(x)$ в точке x_{n+1}, перенесем это условие с правого конца отрезка на левый, выразив

$$y'(x_1) = \frac{y(x_{n+1}) - y(x_1) - \left[S^2\overline{y}^{(2)}\right]_{n+1}}{[SI]_{n+1}},$$

где $\left[S^2\overline{y}^{(2)}\right]_{n+1}$ и $[SI]_{n+1}$ есть $(n+1)$-е компоненты векторов $S^2\overline{y}^{(2)}$ и SI соответственно. Используя формулу Симпсона, получаем

$$\int_0^l \sqrt{1+y'^2}dx \approx \frac{l}{3n}\sum_{i=1}^{n/2}\left(F(y_{2i-1}^{(1)}) + 4F(y_{2i}^{(1)}) + F(y_{2i+1}^{(1)})\right).$$

Таким образом, окончательно приходим к нелинейной системе из $n+1$ уравнений относительно $n+1$ переменной $y_1^{(2)}, \ldots, y_{n+1}^{(2)}$:

$$\begin{cases} \frac{y_j^{(2)}}{1+\left(y_j^{(1)}\right)^2}\left[\alpha\Phi(\overline{y}^{(1)}) + \beta(x_j, y_j)\right] + y_j^{(1)}\beta_x(x_j, y_j) - \beta_y(x_j, y_j) = 0, \\ j = 1, \ldots, n+1, \end{cases}$$

где

$$\begin{aligned} y'(x_1) &= \frac{y(x_{n+1}) - y(x_1) - \left[S^2\overline{y}^{(2)}\right]_{n+1}}{[SI]_{n+1}}, \\ \overline{y}^{(1)} &= Iy'(x_1) + S\overline{y}^{(2)}, \\ \overline{y} &= Iy(x_1) + SIy'(x_1) + S^2\overline{y}^{(2)}, \\ \Phi(\overline{y}^{(1)}) &= \frac{l}{3n}\sum_{i=1}^{n/2}\left(F(y_{2i-1}^{(1)}) + 4F(y_{2i}^{(1)}) + F(y_{2i+1}^{(1)})\right). \end{aligned}$$

4. Проекционные методы. Рассмотрим оператор

$$L(y) = \frac{y''}{1+y'^2}\left(\alpha\int_0^l \sqrt{1+y'^2}dx + \beta(x,y)\right) + y'\frac{\partial\beta(x,y)}{\partial x} - \frac{\partial\beta(x,y)}{\partial y},$$

описывающий полученное необходимое условие минимума для функционала (2):

$$L(y) = 0.$$

Будем искать решение в виде

$$y = \sum_{k=1}^{\infty} a_k\phi_k(x), \tag{4}$$

Здесь $\{\phi_k(x) \mid k = 1, 2, \ldots\}$ – полная система функций в пространстве $\mathbb{C}_0^2[0, l]$ дважды непрерывно-дифференцируемых, финитных на отрезке $[0, l]$ функций. Можно, например, использовать функции

$$\phi_k(x) = \sin\frac{k\pi x}{l}, \quad k = 1, 2, \ldots$$

или

$$\phi_k(x) = (l-x)x^k, \quad k = 1, 2, \ldots$$

Функция y_* удовлетворяет необходимому условию тогда и только тогда, когда $L(y_*)$ ортогональна всем функциям системы $\phi_k(x)$, $k = 1, 2, \ldots$. Рассматривая сумму лишь первых m членов ряда (4), получаем n условий ортогональности

$$\int_0^l L(y_*(x))\phi_k(x)dx = \int_0^l L\left(\frac{y_l}{l}x + \sum_{j=1}^{n} a_j\phi_j(x)\right)\phi_k(x)dx = 0,$$
$$k = 1, \ldots, m.$$

Из полученной системы находим коэффициенты разложения a_j, $j = 1, \ldots, m$, которое определяет приближенное решение задачи.

В методе Ритца функция y_* ищется в том же виде (4), однако здесь это представление подставляется непосредственно в сам минимизируемый функционал стоимости J, а далее определяются значения a_j, $j = 1, \ldots, n$, которые доставляют минимум полученному выражению.

Оба метода сводят исходную бесконечномерную оптимизационную задачу к конечномерной [8, 9].

5. Метод конечных разностей. Необходимое условие минимума можно переписать в виде

$$\begin{cases} y'' = f(x, y, y'), \\ y(0) = 0,\ y(l) = 0, \end{cases}$$

где

$$f(x, y, y') = -\frac{(1 + y'^2)\,(y'\beta_x(x, y) - \beta_y(x, y))}{\alpha\displaystyle\int_0^l \sqrt{1 + y'^2}dx + \beta(x, y)}.$$

Пусть имеется приближенное решение $y_n(x)$. Тогда в его окрестности имеет место разложение

$$f(x, y, y') \approx f\Big(x, y_n, y'_n\Big) + \frac{\partial f}{\partial y}\Big(x, y_n, y'_n\Big)(y - y_n) + \frac{\partial f}{\partial y'}\Big(x, y_n, y'_n\Big)(y' - y'_n).$$

Линеаризованная граничная задача имеет вид

$$\begin{cases} y'' = \psi(x, y_n, y'_n) + f'_{y'}(x, y_n, y'_n)y' + f'_y(x, y_n, y'_n)y, \\ y(0) = 0,\ y(l) = 0, \end{cases}$$

где

$$\psi(x, y_n, y'_n) = f(x, y_n, y'_n) - f'_{y'}(x, y_n, y'_n)y'_n - f'_y(x, y_n, y'_n)y_n.$$

К полученной граничной задаче применим метод пристрелки [10]. Выберем произвольное a в качестве значения для $y'(0)$. Пусть $y_{pt}(x)$ есть решение задачи с начальными условиями

$$\begin{cases} y'' = \psi(x, y_n, y'_n) + f'_y(x, y_n, y'_n)y' + f_y(x, y_n, y'_n)y, \\ y(0) = 0,\ y'(0) = a, \end{cases}$$

и $y_{hm}(x)$ является решением соответствующей однородной задачи

$$\begin{cases} y'' - f'_y(x, y_n, y'_n)y' - f'_y(x, y_n, y'_n)y = 0, \\ y(0) = 0,\ y'(0) = a, \end{cases}$$

Следующее приближение к решению определяется как

$$y_{n+1}(x) = cy_{hm}(x) + y_{pt}(x),$$

где

$$c = -\frac{y_{pt}(l)}{y_{hm}(l)}.$$

В качестве критерия остановки можно выбрать проверку условия

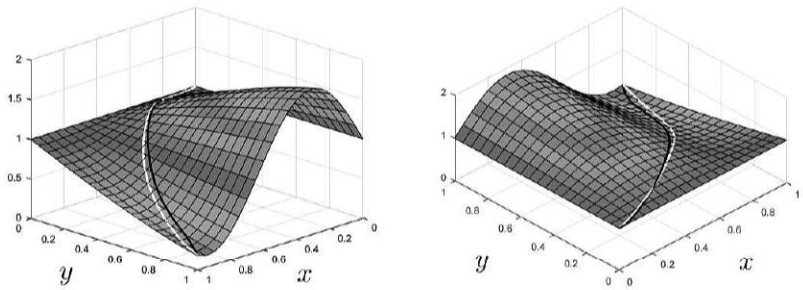
$$|J(y_{n+1}) - J(y_n)| \leqslant \varepsilon$$

при малом $\varepsilon > 0$.

6. Численные эксперименты. Рассмотрим применение описанных в работе алгоритмов к решению следующей задачи: $\alpha = 0{,}1$, $l = y(l) = 1$, $\beta(x, y) = 1 + \sin 5x \sin y$. Для удобства восприятия результатов предположим, что функция $\beta(x, y)$ определяет поверхность рельефа, т. е. чем больше стоимость строительства в точке, тем эта точка выше.

Полученные траектории изображены на рисунках 1 и 2. Результаты вычислений представлены в таблицах 1 и 2. Здесь n – степень

полинома, аппроксимирующего вторую производную искомой функции, m – количество слагаемых в разложении в ряд Фурье, time – время расчетов в секундах, N – количество узлов сетки, iter – количество итераций, требуемых для получения решения, J – значение функционала.



а) б)

Рис. 1. Траектории, полученные методом Галеркина (белая кривая) имеет стоимость 1,279, методом полиномиальной аппроксимации (черная кривая) стоимость – 1,325

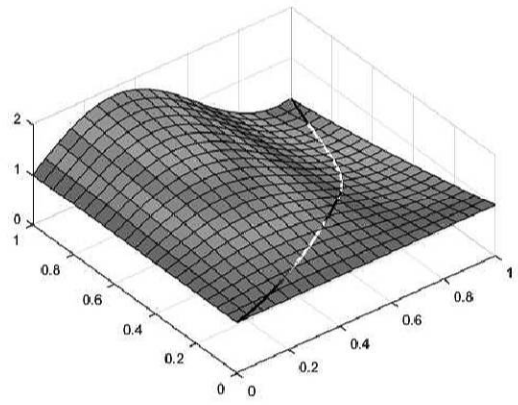


Рис. 2. Траектории, полученные методом Галеркина (белая кривая) и методом конечных разностей (черная кривая)

Таблица 1. Результаты численных экспериментов

Полиномиальная аппроксимация			Метод Галеркина		
J	n	time	J	m	time
1,35403	15	0,6	1,28081	3	0,3
1,34152	19	0,6	1,27881	10	0,7
1,32504	23	0,7	1,27824	30	7

Таблица 2. Результаты численных экспериментов

Метод конечных разностей			
J	N	time	iter
1,27485	100	0,05	4
1,27824	500	0,06	3
1,27824	1000	0,07	3

7. Заключение. Разработанная математическая модель позволяет при сделанных допущениях быстро находить оптимальную по стоимости траекторию, соединяющую две точки на рельефе местности, с любой требуемой точностью. Выведено необходимое условие минимума функционала стоимости, имеющее вид интегро-дифференциального уравнения. При некоторых дополнительных предположениях доказано существование и единственность решения этого уравнения.

Метод полиномиальной аппроксимации обеспечивает решение быстро, однако применим лишь до степеней полинома не больших 25. Попытки увеличить степень для получения более точных решений приводили к численной неустойчивости.

Методы Ритца и Галеркина для рассматриваемой задачи способны выдавать решение с любой точностью, однако требуют существенных вычислительных и, как следствие, временных затрат.

Метод конечных разностей, построенный как комбинация линеаризации и метода пристрелки, решает задачу одновременно и быстро, и точно.

Литература

1. Аббасов М. Э., Шарлай А. С. Поиск оптимальной по стоимости строительства траектории дороги на рельефе местности // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2021. Т. 17. № 1. С. 4–12.

2. Эльсгольц Л. Э. Дифференциальные уравнения и вариационное исчисление. М.: Наука, 1969. 424 с.

3. Люстерник Л. А., Лаврентьев М. А. Курс вариационного исчисления. М.: Государственное объединённое научно-техническое издательство, 1938. 192 с.

4. Люстерник Л. А., Соболев В. И. Элементы функционального анализа. М.: Наука, 1965. 520 с.

5. Треногин В. А. Функциональный анализ. М.: Наука, 1980. 495 с.

6. Abbasov M. E., Sharlay A. S., Variational approach for finding the cost-optimal trajectory // Math Models Comput Simul. 2024. Vol. 16. P. 293–301.

7. Бандурин Н. Г., Гуреева Н. А. Метод и пакет программ для численного решения систем существенно нелинейных обыкновенных интегро-дифференциально-алгебраических уравнений // Математическое моделирование. 2012. Т. 24. № 2. С. 3–16.

8. Канторович Л. В., Крылов В. И. Приближенные методы высшего анализа. Л.: Физматиз, 1962. 708 с.

9. Аббасов М. Э., Шарлай А. С. Метод получения оптимальной по стоимости строительства траектории дороги на рельефе местности // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2023. Т. 19. № 2. С. 139–147.

10. Бахвалов Н. С., Жидков Н. П., Кобельков Г. М. Численные методы. М.: БИНОМ, Лаборатория знаний, 2008. 636 с.

УДК 519.683, 004.4

Ващенков К. В., Пономарев Ю. Ю.

Платформа компании XSQUARE-LCDP – российская Postgres-ориентированная платформа для low code систем – путь в будущее для начинающих разработчиков. Краткое введение

Полное замещение Oracle Forms, Oracle Apex с Pangolin и XSQUARE-LCDP [1]. Данная статья написана в соавторстве с Константином Ващенковым – техническим директором компании «Хи-квадрат». Он является лидером в своей команде разработчиков, которая стала единственной в России и мире, построившей продукты low code разработки приложений, основываясь только на PostgeSQL.

С чего все начиналось. Важно поделиться опытом о том, как создавалась наша платформа, как мы к этому пришли, что делаем сейчас и какой продукт предлагаем на рынок.

Как многие, мы, наверное, половину своей жизни посвятили продуктам иностранных производителей. Однако уже в 2012-14 гг. появились проблемы с их производительностью. Возникла реальная потребность обрабатывать запросы или обращения систем не в сотни миллисекунд, а в десятки и единицы миллисекунд. И это не общие слова, понятно, что нужно ускоряться, замещать множество продуктов и переходить на отечественное, российское оборудование. Не хотелось бы называть это импортозамещением, это на самом деле неинтересно. Было важно просто переходить на свое.

Структура типового трехзвенного приложения и команда его разработки. Давайте посмотрим на базовую архитектуру приложения. Ничего нового: сервер баз данных, сервер приложений, фактически де-факто стандарт, при этом используется балансировщик нагрузки в виде Apache, Nginx и серверы приложений.

Ващенков Константин Вячеславович – технический директор, ООО «Хи-квадрат»; e-mail: kv@xsquare.ru, тел.: +7(915)472-31-49

Пономарев Юрий Юрьевич – эксперт, ООО «Автоматизированные системы, консалтинг, информационные технологии»; e-mail: yura_graph@mail.ru, тел.: +7(903)537-46-11

По классике, разработку ведут три программиста: по СУБД, по среднему слою, по интерфейсам пользователя. Если по какому-то конкретному проекту необходимо обеспечить быстрый рост разработки, то команда увеличивается. Тогда минимальная команда – Dream Team, как и у любого среднего проекта, возрастает на три человека по каждому направлению.

Итак, десять человек – это уже приличный бюджет, появляется риск нехватки ресурсов, что может стать катастрофой для проекта. Именно поэтому, сейчас многие готовы поставлять платформы, а внедрять их не готовы, так как нет ресурсов для внедрения, для предоставления количественного скачка их конечному бизнес-приложению.

Как мы экономили на знаниях и надеялись на Oracle. Что в этом случае делать? Давайте не будем ничего выдумывать, а для начала сделаем обзор рынка – простой, маленький, далеко ходить не будем. Можно сказать, что в 90-е – 2000-е Oracle занял монополию по рынку разработки. Каким образом? Oracle выпустил четыре продукта, а именно: Oracle Apex, Oracle Forms, среду разработки, REST API – RDS, скомпоновав его либо в PL/SQL mode, либо в Report Server.

На что мы купились? Мы купились на две вещи. Во-первых, чтобы на всем этом разрабатывать, именно на этом технологическом стеке, нужен только один программист, который должен знать два языка СУБД: SQL и PL/SQL. Таким образом, студент третьего курса университета в течение полугодовой-годовой практики готов «лепить» относительно промышленный код. Да, он может быть нестройным, некрасивым, но это все же полностью готовый разработчик и всего лишь один. Он может заменить всех! Во-вторых – это Oracle Database, которая прощает все ошибки, т. е. можно писать неоптимальные соединения, неоптимальные запросы... Можно даже декартово произведение сделать, но в какой-то момент Oracle прощал все.

Простая и действенная замена СУБД на отечественную. А давайте слово Oracle заменим на Postgres, ну казалось бы, просто одно слово и Oracle нет, есть Postgres. Что нам для этого нужно сделать? Да те же самые четыре базовых продукта. Первый – это сервер приложений (PGHS [2,3]), второй – среда разработки (XRAD [3]), третий – RESTful API Server (XDAC [4]) и четвертый – сервер

отчетов (XREPORT [5]). Очевиден тот же тезис – нам нужен только один SQL [6], PL/pgSQL [7] разработчик и все... Да! – и сам Postgres. Для разработчиков отчетности рекомендуем посмотреть на стандарт JSON [8].

Десять лет ради четырех продуктов – наш вклад в информационные технологии России. Такими вопросами мы задались порядка десяти лет назад, и порядка пяти лет назад представили рынку платформы Хи-квадрат (Xsquare), которые умеют делать все то же самое на платформе Postgres.

Говоря про разработку, условно как внутренний сервер после Apache, у нас на платформе нет legacy, у нас нет исторического долга, мы можем сделать ядро такое, как нам удобно, которое считаем нужным. Мы не зависим ни от какого стороннего продукта, не являемся его форком или последователем какого-то open source продукта. Все у нас самостоятельнее, все с нуля, поскольку целью было решить все те базовые проблемы, которые не мог решить Oracle Application в течение трех десятилетий.

Немного о внутренностях предлагаемых решений. Первое, что требуется от приложения, что действительно важно – это его скорость. Что нам обеспечивает скорость? Это компилируемые языки программирования, соответственно в нашей платформе отсутствуют Java, C#, Python. Ничего ресурсоемкого нет, чистый C++ и Go. Мы должны показывать супер производительность на относительно простом железе. Это гарантируется использованием базовых языков и независимостью от внешних решений. Другими словами, мы в состоянии взять платформу, перенести на любую другую базовую архитектуру, откомпилировать и готово.

Таким образом платформа абсолютно нормально работает как на x86, так и на ARMax, Baikalax и Эльбрусах. Например интересно, что маленький двухядерный Эльбрус у нас показывает производительность порядка 20–30 запросов в секунду, т. е. на нем может нормально работать порядка 100 пользователей. А на маленьком NetTop, если взять большие серверы, двухсокетные, односокетные, то там уже сотни запросов и на нем могут работать тысячи пользователей.

Как внутренности предлагаемых решений сокращают время и пространство. Итак, про архитектуру: главное достоинство приложения в первую очередь – это его скорость. Однако иногда

нам этого мало, соответственно, посредством реверс прокси-серверов можно сделать балансировку насколько это удобно. Из практического опыта у нас больше четырех узлов никогда не было, потому что даже двух узлов хватает для большинства практических задач. Наконец, это два независимых веб-контроллера, один для сервера, а другой для бизнес-приложения.

Вспомним, как работает приложение Oracle. Обычно, оно работает как нечто монолитное, единое, дающее тесную интеграцию данных, но лишающее разработчика возможности масштабирования. Соответственно, первое, что было сделано – мы порвали связи на несколько независимых частей: разделили метаданные на саму базу данных, где хранится информация о нашем приложении, и собственно само приложение. Таким образом, сделали так, что есть место, где хранятся ваши клиенты, договоры и так далее. А есть место, где работают команды, которые отвечают за разработку, и другие, которые отвечают за администрирование и сопровождение.

Несколько слов о рекомендуемой команде разработки. Давайте пройдемся по ключевым показателям. Как ранее было сказано, необходимо знать только Platform V Pangolin. Что это дает, если у вас есть задача сделать какую-то MVP? Для вас необходим всего лишь один SQL-разработчик. Это может быть разработчик Oracle, это может быть разработчик с навыками SQL Server Microsoft, с MySQL, с любой базой. Из нашего опыта, для того, чтобы разрабатывать MVP с квалификацией Oracle Developer достаточно два-пять дней подготовки, Microsoft SQL Developer – один-два дня.

Как нам сказал один из заказчиков: «Вы создаете нам нашу маржинальность, вы позволяете нам зарабатывать деньги». У этого заказчика было приложение на несколько тысяч страниц – это Forms или Apex. Чтобы писать его на Java и C#, требовалась команда в 30 человек. Разработчики просто не успевали нанимать специалистов и по бюджету не укладывались. В результате, по нашей рекомендации поискали на HeadHunter бывших SQL разработчиков, что буквально за одну-две недели позволило закрыть вопрос. Их быстро подготовили, организовали в одну производственную линию, соответственно, проект закрыли в полгода, а бюджет фонда оплаты труда на разработку сократился на 80%. За счёт того, что нет дорогих Java-девелоперов, нет дорогих фронттендеров, этой компании был необходим только самый дешевый и популярный разработчик – это

разработчик SQL, PL/pgSQL, что в итоге значительно сэкономило бюджет.

Ахиллесовы пяты готовых решений и как их обрабатывает платформа LCDP. Теперь обсудим две основные проблемы, которые Oracle не мог решить за 30 лет, а мы решили, потому что у нас была возможность самостоятельно писать ядро.

Представьте, что у нас несколько экземпляров базы данных, Platform V Pangolin. Если взять любое приложение на runtime Oracle, то оно монолитно связано с Oracle. Да, можно делать микросервисную архитектуру, но это долго и дорого. Безусловно, местами это необходимо, более того, начиная с 20-х годов, на Западе активно муссируется тезис, что монолитное приложение не есть плохо. Да, для большинства продуктов, проектов монолитное приложение есть большая польза и выгода, но почему?

Например, задача сделать приложение, которое консистентно, быстро разрабатываемо и быстро внедряемо. Что для этого сделали мы? Была реализована возможность из ядра системы подключать одновременно неограниченное количество источников данных, т. е. вы делаете огромное приложение на тысячу или десять тысяч страниц, ровно так, как вам удобно, как вы привыкли это делать. Почему такое решение возможно? Потому, что само приложение кластеризуется сразу из коробки нашей платформы XSQUARE-LCDP.

А что же делать с тем, что находится в бэкенде? Это делается очень просто. Предположим, у вас есть большая монолитная база на несколько тысяч таблиц, вы логически разделяете ее либо шардируете на несколько экземпляров. Допустим в одной базе – клиенты, в другой – заявки, в третьей – транзакции, в четвертой – договоры. Безусловно, из опыта, под самые высоконагружные источники и под самую важную информацию, такую, как транзакции и договоры, вы выделите лицензии платной отечественной базы данных. Это важно хотя бы для безопасности данных. А уже все второстепенные данные, легко закрываются, в том числе, ванильной СУБД.

Что мы таким образом получаем от платформы XSQUARE-LCDP? Платформа позволяет разработчику, при сохранении принципов монолитной разработки, брать любой элемент, любой контрол и подключать его к любому источнику данных, т. е. зайдя на условную страницу форы, например, клиента, у вас появляется возможность обращения к заявкам, договорам и т. п. Легко и далее шарди-

ровать ваше приложение. Соответственно, платформа автоматически забирает данные из того источника, из которого нужно. И все... Таким образом, разрабатывается монолит, а в бэкэнде он остается горизонтально масштабируемым.

Следующая проблема клиентов – это рендеринг на сторонних вебсервисах. Мы видели достаточно много такого кода, поэтому сервис-рендеринг – это, во-первых, дорого, во-вторых, задействует ресурсы серверного оборудования и все это, чаще всего, происходит как дополнительная нагрузка в рамках ресурсов корпоративной сети.

Мы же реализуем специализированный веб-контрол для построения такого функционала, протокол обмена с которым оптимизирован настолько, что, даже включая обмен JSON-файлами, для рендеринга формы ему нужно два-десять-двадцать килобайт трафика для отправки типичного запроса к базе данных до пяти килобайт в целом, создавая результативный трафик в один-два килобайта. Если мы, к примеру, ищем клиента по ИНН, то объем трафика для поиска составит 250–500 байт.

Не килобайт, не мегабайт, а всего лишь байт. Когда мы говорим про встроенный Connection Pool, да, безусловно, можно использовать любые балансеры, другие сторонние решения, но в базовой инсталляции платформа предоставляет отличный пулер.

Таким образом, на текущий момент, при текущих показателях производительности, мы можем сказать, что если вы правильно пишете приложение, у вас хорошая селективность данных, хорошо построенная архитектура, то у нас одна сессия базы данных в вполне в состоянии обрабатывать сотни запросов TPS. Время отклика СУБД на оборудовании Intel 2020-23 гг. составляет примерно 7 мс.

Почему мы не используем микросервисы. Клиенты часто приходят с запросом: «Дайте нам микросервисы! Желательно так, как мы делали их раньше».

Наш продукт использует внутренний HTTP-клиент. В Postges – это HTTP-клиент, устанавливаемый в виде расширения. В итоге, неправильное пожелание клиента вынуждает нас тратить самый дорогой ресурс – ресурс базы данных. Поэтому мы реализовали полный, полноценный HTTP-клиент в серверных приложениях. В результате, серверные приложения, минуя базу данных, минуя любые третьи источники, полностью самостоятельно могут обращаться к любым третьим источникам. Таким образом, вы можете масштаби-

ровать серверные приложения неограниченно, вы можете полностью построить приложение целиком на микросервисах, если вам это нужно.

Говоря про производительность, приведем пример крупнейшего заказчика – ВТБ, более 11 миллионов клиентов; заказчик РЖД – на несколько миллионов клиентов. В промышленной среде приложение занимает порядка 150–200 мегабайт трафика и совсем немного ядер. Его ресурсы вполне сравнимы с таким же по сложности сервером на Apache. Время отклика при этом достигает 63 мс – обычный хороший отклик.

Интерфейс и будущее среды разработки. Безусловно, интерфейс любого приложения важен. Это как раз та проблема, которую Oracle не мог решить годами. У нас были значительные инвестиции, соответственно, вы сразу же получаете интерфейс, который адаптивен для всех устройств, сразу получаете готовые приложения, один раз – и готово. Для тех, кто полюбил Oracle Apex, у нас есть скин, который фактически для разработчика повторяет скин Oracle Apex.

Хочется сказать немножко вперед, в конце этого года (речь идет о планах на 2024 год, они реализовались в текущем, 2025 году) у нас выходит пятая версия платформы XSQUARE-LCDP. Это будет среда разработка на стероидах, где будет что-то среднее между Net Beans, Visual Studio и текущей средой разработки, которая позволит ускорить разработку от текущего версии раза в три. Помните? Нам важна ваша маржинальность. Сейчас вы и так все быстро разрабатываете, но мы хотим, чтобы вы разрабатывали еще быстрее.

О двух не менее важных продуктах – XDAC и XREPORT. Первый продукт – XDAC – это RESTful API-сервис для публикации сервисов доступа к данным, а второй – XREPORT – это сервер отчетов.

Сервер отчетов решает две базовые задачи. Первая – это генерация DOCX-XLSX-PDF – бинарно-нативно, вторая – в обратную сторону, он их парсит-нативно.

Как вы ранее заметили, у нас есть здоровая паранойя на производительности. Соответственно, скорость формирования XLSX на 10000 строк – одна секунда. Если мы говорим про один документ, это от одной до десяти миллисекунд.

Теперь у нас есть нативный язык разработки и его базовые биб-

лиотеки. На 100% кода решение реализовано самостоятельно. Мы независимы в рамках продуктов. Все без opensourse. Что это дает? Мы в состоянии практически под любой запрос заказчика реализовать его потребность. Почему? Потому что у нас есть компетенция.

Поговорим про XDAC – RESTful API-сервис. Здесь наша цель была достаточно проста, превратить наш Postgres в RESTful API-сервис, чтобы ускорить публикацию отчетов. Фактически, берем один SQL запрос. Оборачиваем его в XDAC, и у вас за 5–10 минут готов микросервис. Фабрику из десятка, даже сотни микросервисов вы в состоянии написать за неделю, одним разработчиком. Производительность аналогичная, 15 мегабайт оперативной памяти, максимум 100 мегабайт трафика для тысяч клиентов. Соответственно, есть кэш запросов, двоичный для получения отчета в локальную директорию или BASE-64 кодирования для передачи отчета через Интернет.

Ответ на популярный вопрос: «Будет ли создан конвертер с западных форм на продукты Хи-квадрат?». Вопрос о создании конвертера, например, для Apex поднимается практически всегда. Это обычно способ перенести формы, если у кого-то есть разработки. Для Oracle Forms это будет бесполезно, потому что выглядит ужасно.

Начнем с небольшой истории. По поводу Oracle Forms: если вы возьмете любое текущее современное решение, то ни одно из них при всей ужасности Oracle Forms не предоставляет того объема функционала и работы, которое предоставляет Oracle Forms, который был реализован 20–30 лет назад, а то и больше. И все, куда сейчас идет текущая мировая разработка, это упрощение Oracle Forms, т. е. все идеи были заложены тогда! Как следствие, если в Oracle Forms тогда было порядка 150–200 триггеров на уровне формы, на уровне runtime, на уровне datablock, то в текущих средствах разработки они просто сведены к пяти – десяти. Аналогично с Oracle Buisiness Suite, там существует, 6–8 триггеров в общем значении, где можно модифицировать свое ПО. Oracle пытался сделать конвертеры с Forms Apex и у самой корпорации ничего не получилось.

Дальше, когда мы говорим про Oracle Apex, то базовый конвертер есть, но он достаточно примитивный, он позволяет создавать страницы, регионы и так далее. В результате 100% заказчиков, подчеркнем, не 95%, не 99%, а 100% заказчиков, сразу говорят: дайте конвертер, они даже не спрашивают, что умеет продукт, что не умеет

продукт, просто дайте конвертер, чтобы мигрировать.

Спрашиваешь: а для базы данных есть на рынке хотя бы один конвертер, который конвертирует базу один в один? Оказывается – нет, т. е. можно много услышать, что есть, но он работает процентов на 30, это больше помощник такой... А дальше, перенесли как-то базу и на этапе миграции все начинают понимать, что здесь уже проблемы, а в чулане зарыто что-то еще, что там куча легаси и того, что уже не используется... В результате 100% заказчиков решают, что им нужна не миграция, а переходы с глубоким рефакторингом, с сохранением команды разработки.

Все разработчики, которые 15 лет сидят над приложением, которые делали какую-то банковскую систему, учетную систему и так далее – это уже не разработчики, это бизнес-подразделение, которое является носителем прикладного функционала.

Наша главная цель – не миграция приложения, а здравый рефакторинг, переделка приложения с переносом и сохранение команды разработчиков. Это работает для всех заказчиков, при этом задача несколько трансформируется. И с каждым заказчиком мы готовы работать и воспроизводить его потребности нашими возможностями и нашими продуктами.

Литература

1. XSQUARE-LCDP (Low Code Development Platform) [Электронный ресурс]: URL:https://www.tadviser.ru/index.php/%D0%9F%D1%80%D0%BE%D0%B4%D1%83%D0%BA%D1%82:XSQUARE-LCDP_(Low_Code_Development_Platform) (дата обращения: 01.05.25).

2. PGHS + RAD. Лоу код платформа для корпоративных приложений [Электронный ресурс]: URL:https://lcdp.xsquare.ru/ (дата обращения: 01.05.25).

3. Руководство разработчика XSQUARE – XRAD 5.1.2 [Электронный ресурс]: URL:https://lcdp.xsquare.ru/docs/rad/5_0/ (дата обращения: 01.05.25).

4. XDAC – Data Access Connector [Электронный ресурс]: URL: https://lcdp.xsquare.ru/docs/xdac/3_4/ (дата обращения: 01.05.25).

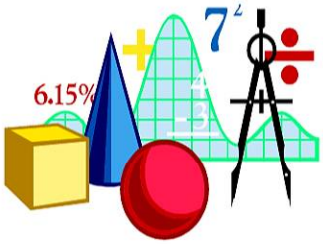
5. XSQUARE – REPORTS [Электронный ресурс]: URL:https://lcdp.xsquare.ru/docs/xreports/5_0/ (дата обращения: 01.05.25).

6. Документация к PostgreSQL 17.5 [Электронный ресурс]: URL:https://postgrespro.ru/docs/postgresql/current/sql (дата обращения: 01.05.25).

7. Chapter 41. PL/pgSQL — SQL Procedural Language [Электронный ресурс]: URL:https://www.postgresql.org/docs/current/plpgsql.html (дата обращения: 01.05.25).

8. Стандарт JSON [Электронный ресурс]: URL:https://www.json.org/json-ru.html (дата обращения: 01.05.25).

1. Математическая теория процессов управления



УДК 517.929.4

Евтина Д. С., Жабко А. П.

Оценка перерегулирования уравнений нейтрального типа с несоизмеримыми запаздываниями

1. Введение. Среди различных видов систем дифференциальных уравнений с запаздывающим аргументом [1–4] в настоящее время большое пространство для исследований остаётся среди уравнений нейтрального типа. На практике такие уравнения возникают в механических системах с задержкой информации. Случай одного запаздывания в нейтральной части рассмотрен в работе [5]. Известно также [5], что когда запаздывания в нейтральной части соизмеримы, то систему можно свести к виду одного запаздывания. Управляемые системы характеризуются двумя важными параметрами: запасом устойчивости и перерегулированием. Целью работы является разработка подхода к оценке решений уравнения с двумя несоизмеримыми запаздываниями в нейтральной части, позволяющей найти величину перерегулирования.

2. Постановка задачи. Рассмотрим скалярное дифференциально-разностное уравнение с двумя несоизмеримыми запаздываниями

$$\dot{x}(t)-d_1\dot{x}(t-\tau)-d_2\dot{x}(t-h) = a_0x(t)+a_1x(t-h)+\int\limits_{-\tau}^{0} q(\theta)x(t+\theta)d\theta, \quad (1)$$

где a_0, a_1, d_1, d_2 – заданные отличные от нуля коэффициенты, а $q(\theta)$ – ограниченная кусочно-постоянная функция. Положим $\tau \geqslant h$ и соотношение запаздываний $\frac{\tau}{h}$ как иррациональное число. Далее предполагаем, что уравнение (1) экспоненциально устойчиво [1].

Поскольку рассматриваемое уравнение является стационарным, будем считать начальный момент времени $t_0 = 0$. Тогда решение уравнения (1) определяется начальной функцией $\varphi \in \mathcal{C}^1([-\tau, 0], \mathbb{R})$: $x(\theta) = \varphi(\theta),\ \theta \in [-\tau, 0]$.

Евтина Диана Сергеевна – аспирант, Санкт-Петербургский государственный университет; e-mail: diana.evtina@mail.ru, тел.: +7(981)883-27-89

Жабко Алексей Петрович – профессор, Санкт-Петербургский государственный университет; e-mail: zhabko.apmath.spbu@mail.ru, тел.: +7(911)779-67-91

Задача состоит в построении оценки решений уравнения (1).

3. Предварительные сведения. Сформулируем несколько определений, необходимых для решения поставленной задачи.

Определение. Величина $k(t)$ называется фундаментальной матрицей уравнения (1), если удовлетворяет следующим условиям [1]:
1)

$$k(t) = \begin{cases} 0, & t < 0, \\ 1, & t = 0; \end{cases}$$

2)

$$\frac{d}{dt}\Big[k(t) - d_1 k(t-\tau) - d_2 k(t-h)\Big] = a_0 k(t) + a_1 k(t-h) + \\ + \int\limits_{-\tau}^{0} q(\theta) k(t+\theta) d\theta, \ t > 0, \ t \neq m\tau + lh;$$

3) $\Big[k(t) - d_1 k(t-\tau) - d_2 k(t-h)\Big]$ непрерывна при $t \geqslant 0$, т. е.

$$k(t+0) - d_1 k(t-\tau+0) - d_2 k(t-h+0) = \\ = k(t-0) - d_1 k(t-\tau-0) - d_2 k(t-h-0).$$

При этом $k(t)$ претерпевает разрывы в точках $m\tau + lh$, где m и l – целые положительные числа.

Решение уравнения (1) может быть представлено в форме Коши [1]:

$$x(t,\varphi) = \Big[k(t) - d_1 k(t-\tau) - d_2 k(t-h)\Big]\varphi(0) + \\ + d_1 \int\limits_{-\tau}^{0} k(t-\tau-\theta)\varphi'(\theta)d\theta + \int\limits_{-h}^{0} k(t-h-\theta)\Big[a_1\varphi(\theta) + d_2\varphi'(\theta)\Big]d\theta + \\ + \int\limits_{-\tau}^{0}\Big(\int\limits_{-\tau}^{\theta} k(t-\theta+\xi)q(\xi)d\xi\Big)\varphi(\theta)d\theta.$$

4. Основной результат. Известно [6], что если уравнение (1) экспоненциально устойчиво, то необходимо выполнение условия $|d_1| + |d_2| < 1$.

Перепишем уравнение (1) в интегральной форме:

$$x(t) = d_1 x(t-\tau) + d_2 x(t-h) + f(t), \tag{2}$$

где

$$f(t) = x(0) - d_1 x(-\tau) - d_2 x(-h) + a_0 \int_0^t x(\xi)d\xi + a_1 \int_0^t x(\xi - h)d\xi +$$

$$+ \int_0^t \int_{-\tau}^0 q(\theta)x(\xi+\theta)d\theta d\xi.$$

Введём оператор запаздывания $e^{-pr} \circ f(t) = f(t-r)$. Тогда множество

$$K = \Big\{ ae^{-ph} + be^{-p\tau} : a, b \in \mathbb{R},$$

$$(ae^{-ph} + be^{-p\tau}) \circ f(t) = af(t-h) + bf(t-\tau) \Big\}$$

образует коммутативное кольцо операторов над полем вещественных чисел.

Применяя оператор запаздывания к правой части уравнения (2) $k-1$ раз, получим

$$x(t) = (d_1 e^{-p\tau} + d_2 e^{-ph})^k \circ x(t) + \sum_{j=1}^{k} f_j(t),$$

где $f_1(t) = f(t)$, а $f_j(t),\ j = \overline{2,k}$, задаются рекурсивно:

$$f_j(t) = d_1 f_{j-1}(t-\tau) + d_2 f_{j-1}(t-h).$$

Так как требуется выразить решение через начальную функцию, то рассмотрим значение k, удовлетворяющее неравенству

$$t - \tau \leqslant m\tau + (k-m)h \leqslant t, \quad m = \overline{0,k}. \tag{3}$$

Из неравенства (3) вытекают следующие оценки:

$$\frac{t}{\tau} - 1 \leqslant k_1 \leqslant k \leqslant \left[\frac{t}{h}\right] = k_2,$$

$$k \leqslant m \leqslant k + \left[\frac{t-kh}{\tau-h}\right] \leqslant k + \left[\frac{\tau}{h}\right], \quad \left[\frac{\tau}{h}\right] = \overline{m}.$$

Тогда справедлива

Лемма. *Решение уравнения (1) представимо в виде*

$$x(t) = \sum_{k=k_1}^{k_2} \Bigg(\sum_{m=0}^{\overline{m}} C_k^m d_1^m d_2^{k-m} \varphi\big(t - m\tau - (k-m)h\big) + \\ + \sum_{m=0}^{\overline{m}} C_{k-1}^m d_1^m d_2^{k-1-m} f_1\big(t - m\tau - (k-1-m)h\big) \Bigg).$$

Следствие. *Решение уравнения (1) допускает оценку*

$$|x(t)| \leqslant \sum_{k=k_1}^{k_2} \Bigg(\sum_{m=0}^{\overline{m}} C_k^m |d_1|^m |d_2|^{k-m} |\varphi\big(t - m\tau - (k-m)h\big)| + |f_k(t)| \Bigg) \leqslant \\ \leqslant \sum_{k=k_1}^{k_2} \Big[\Big(|d_1| + |d_2| \Big)^k |\varphi|_\tau + |f_k(t)| \Big],$$

где $|\varphi|_\tau = \sup\limits_{\theta \in [-\tau, 0]} |\varphi(\theta)|$.

5. Заключение. В данной работе представлен метод оценки решений дифференциальных уравнений с запаздыванием нейтрального типа, проиллюстрированный на примере оценки перерегулирования уравнения с двумя несоизмеримыми запаздываниями в нейтральной части. В дальнейшем полученный результат можно обобщить для систем с несколькими несоизмеримыми запаздываниями в нейтральной части, систем с периодическими коэффициентами, период которых несоизмерим с запаздыванием, а также для разностных систем.

Литература

1. Kharitonov V. L. Time-delay systems: Lyapunov functionals and matrices. Basel: Birkhauser, 2013. 311 p.

2. Эльсгольц Л. Э. Введение в теорию дифференциальных уравнений с отклоняющимся аргументом. М.: Наука, 1964. 128 с.

3. Евтина Д. С. Функционалы Ляпунова – Красовского для однородных уравнений с распределённым запаздыванием // Процессы управления и устойчивость. 2022. Т. 9. № 1. С. 44–48.

4. Хоанг В. Н. Дифференциально-разностная система в соболевском пространстве функций с носителями на сетеподобной области // Процессы управления и устойчивость. 2022. Т. 9. № 1. С. 113–117.

5. Kharitonov V. L. Exponential estimate for a simple neutral time delay system // Course of lectures given in St. Petersburg State University. 2012. 19 p.

6. Евтина Д. С., Жабко А. П. Исследование устойчивости систем дифференциальных уравнений с запаздыванием нейтрального типа // Материалы международной научной конференции «Уфимская осенняя математическая школа». 2024. Т. 2. С. 67–69.

УДК 62-52

Живулин Е. А., Гришкин В. М.

Построение маршрута движения судна с помощью методов минимизации с целью экономного расхода топлива

1. Введение. Существенная часть расходов на содержание судна заключается в затратах на топливо, поэтому оптимизация маршрута [1] с целью экономии расхода топлива является актуальной задачей. Для решения задачи по оптимизации маршрута требуется функция зависимости величины расхода топлива от маршрута судна и его скоростей на нем.

Для построения маршрута и определения его стоимости используется программа, запрашивающая информацию о курсовых углах движения управляемого судна до выбранных движущихся субъектов, которые предполагается обходить на выбранном маршруте, а также скорость во время обхода того или иного субъекта.

2. Программа моделирования движения судна. Судно идет при заданном курсовом угле до тех пор, пока не достигнет заданной дистанции с субъектом, который предполагается обходить, после чего будет совершен обход субъекта со скоростью, величина которой является одной из переменных, задающих параметры движения и также передается в программу расчета и моделирования движения судна [2]. Обход субъекта может происходить как с левой, так и с правой стороны относительно субъекта. Этот параметр также передается в программу.

Параметры передаются в следующем виде: {номер субъекта; угол движения к нему; скорость обхода; сторона обхода}, {номер субъекта; угол движения к нему; скорость обхода; сторона обхода}... Параметры передаются в программу через сокеты.

Для решения задачи поиска экономичного маршрута в том или ином случае расположения и движения субъектов, требуется решить

Живулин Евгений Алексеевич – аспирант, Санкт-Петербургский государственный университет; e-mail: st078725@student.spbu.ru, тел.: +7(981)876-34-42

Гришкин Валерий Михайлович – доцент, Санкт-Петербургский государственный университет; e-mail: v.grishkin@spbu.ru, тел.: +7(921)790-86-59

задачу поиска минимума функции зависимости расхода топлива от заданного набора параметров.

Программа, строящая маршрут судна и определяющая стоимость выбранного маршрута, реализована на языке программирования C#. Стоимость маршрута складывается из стоимости произошедших событий:

1. За столкновения с другими субъектами начисляется большое число баллов, чтобы для методов минимизации избегание столкновений было приоритетной задачей.

2. За выбор таких курсовых углов, при которых потребуется двигаться со скоростью, которую не может обеспечить судно.

3. За построение маршрута, прохождение которого приведет к опозданию к назначенному времени.

4. За расход топлива при движении по маршруту.

3. Поиск оптимального маршрута. Программа по поиску решения реализована на языке программирования Python с использованием функции optimize.minimize библиотеки SciPy.

Для того, чтобы метод поиска минимума функции изменял не только углы движения корабля и значения его скоростей при обходе движущегося субъекта, но и порядок обходимых субъектов, а также сторону обхода, для метода поиска минимума эти параметры представлены в виде чисел, изменение которых приводит к изменению порядка и стороны обхода. После формирования набора параметров в числовом виде, они передаются в функцию, которая конвертирует их в формат программы моделирования.

Порядок обходимых субъектов должен изменяться для разной длины шага метода поиска минимума, иначе есть вероятность неспособности метода изменять порядок обходимых субъектов при достижении определенной длины шага. Поэтому реализован алгоритм изменения номера субъекта, если происходит изменение значения номера обходимого судна. Если значение больше нуля, то номер судна, передаваемый в программу моделирования, увеличивается на единицу, если меньше нуля – уменьшается на единицу. Если номер судна в результате этих операций становится больше максимального номера, то его значение становится равным минимальному номеру, и

наоборот, если значение меньше минимального номера, ему приравнивается максимальный.

При реализации рассмотренного выше подхода для поиска минимума были применены следующие алгоритмы поиска минимума, реализованные в библиотеке SciPy: BFGS [3, 136 с.], CG [3, 459 с.], COBYLA [4, 51 с.], L-BFGS-B [5], Powell [6], Newton-CG [3, 168 с.], trust-krylov [7]. Для определения наиболее подходящего, каждый из этих методов был применен в одних и тех же условиях, т. е. для одного набора задач при одних и тех же изначальных параметрах. Методы были использованы для решения 50 задач. В каждой задаче присутствует 5 движущихся субъектов. Так как результат работы метода зависит от изначального набора параметров, для каждого из 50 случаев было сгенерировано 50 случайных наборов параметров.

В таблице отмечено, что метод решил задачу, если хотя бы за одну из 50 попыток ее решения он нашел набор параметров, который позволяет маневрировать без столкновений и прийти в пункт назначения в пределах заданного времени прибытия, и при этом результаты решения этой задачи другими методами не являются более оптимальными, или отличаются незначительно: менее, чем на 2% от результата.

Далее в таблице приведен результат работы методов, демонстрирующий число решенных задач (максимальное значение 50), число успешных попыток решения задачи (максимальное значение 2500), время решения всего набора задач, среднее время решения задачи при одном заданном наборе изначальных параметров.

По результатам эксперимента очевидны два метода, демонстрирующие лучшие показатели в разных категориях. Метод COBYLA с наибольшим числом успешных попыток решения задач с разным набором изначальных параметров, который при этом решил всего 33 задачи. Метод Powell, у которого 912 успешных попыток, но при этом 44 решенных задачи. Это может означать, что метод COBYLA больше зависим от типа задачи, в то время, как Powel сильнее зависят от набора начальных значений параметров.

В связи с тем, что ни один метод не решил все задачи, и при этом для успешного решения задач требовалось использовать несколько наборов изначальных значений, следует, что для поиска оптимального пути необходимо найти изначальный набор параметров, который позволит решить задачу с помощью метода поиска минимума.

Таблица. Сравнение методов поиска минимума функции (в процессе обработки)

Метод	**Число успешных попыток решения**	**Число решенных задач**	**Время решения задач**	**Среднее время решения задачи**
BFGS	993	31	45 часов 21 минута	1 минута 5 секунд
CG	1011	31	39 часов 17 минут	57 секунд
COBYLA	1555	33	3 часа 6 минут	5 секунд
L-BFGS-B	798	31	15 часов 50 минут	23 секунды
Powell	912	44	35 часов 39 минут	51 секунда
Newton-CG	1500	30	2 часа 9 минут	3 секунды
trust-krylov	1500	30	1 час 53 минуты	3 секунды

Для гарантированного решения задачи с помощью метода поиска минимума функции предлагается находить такой набор параметров, который позволит совершить маневр, удовлетворяющий минимальным требованиям: избегать столкновений с другими субъектами и приводить судно в пункт назначения вовремя. После чего применение метода поиска минимума позволит сделать маневр более выгодным при относительно небольшой коррекции переменных.

4. Заключение. Поиск решения задачи по построению маршрута методом поиска минимума функции стоимости маршрута от заданных параметров позволяет найти решение в большинстве случаев. Процент решенных задач недостаточно большой для того, чтобы можно было использовать рассмотренный способ без доработок и модификаций.

Даже если используемый метод поиска минимума может решить ту или иную задачу, для этого может потребоваться несколько попыток с различными изначальными значениями параметров, что непредсказуемо увеличивает время решения.

Результат работы методов минимизации зависит от изначального значения параметров настолько, что приводит к необходимости предварительно подбирать набор значений, после чего производить

поиск минимума. Тем не менее, данный подход позволяет увеличить точность решения, которое было найдено более грубым способом.

Литература

1. Тазиева Л. М. Программная реализация и модификация алгоритма поиска оптимального пути на графе JPS+ // Процессы управления и устойчивость. 2018. Т. 5. № 1. С. 357–364.

2. Живулин Е. А., Гришкин В. М. Расчет траектории обхода движущегося судна с целью экономии топлива // Процессы управления и устойчивость. 2023. Т. 10. № 1. С. 56–64.

3. Nocedal J., Wright S. J. Numerical Optimization. Second Edition. New York: Springer, 2006. 686 p.

4. Powell M. J. D. Advances in Optimization and Numerical Analysis. First Edition. Oaxaca: Springer, 1994. 294 p.

5. Zhu C., Byrd R. H., Lu P., Nocedal J. Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization // ACM Transactions on Mathematical Software. 1997. No 23. P. 550–660.

6. Arouxet M. B., Echebest N., Pilotta E. A. Active-set strategy in Powell's method for optimization without derivatives // Computational and Applied Mathematics. 2016. No 30. P. 171–196.

7. Gould N., Lucidi S., Roma M., Toint P. Solving the Trust-Region Subproblem using the Lanczos Method // SIAM Journal on Optimization. 1999. No 9. P. 504–525.

УДК 517.977

Каминцева В. С.

Построение оптимального регулятора для систем с возмущениями

Рекомендовано к публикации профессором Смирновым Н. В.

1. Введение. Проблемы оптимального управления активно исследуются в мировой литературе. Результаты нашли свое применение в различных областях науки, техники и экономики. Один из методов синтеза оптимальных управлений разработан Р. Габасовым и представителями его научной школы [1]. В основе его подхода лежит алгоритм сведения динамической задачи оптимального управления к интервальной задаче линейного программирования (ИЗЛП). Этот подход, с некоторыми изменениями и дополнениями в зависимости от специфики задачи и требуемыми целями, применяется для решения множества прикладных проблем [2–9].

В данной статье рассматривается линейная неоднородная управляемая система. Для нее методом Габасова строится управление типа обратной связи при наличии ограниченного возмущения, неизвестного регулятору. Результаты иллюстрируются примером построения оптимальных управлений для электродвигателя.

2. Постановка задачи оптимального управления. Будем следовать работе [1].

2.1. Классическая задача. Функцию $u(t)$, $t \in T = [t_*, t^*]$, назовем *дискретным управлением* с периодом дискретизации h, если $u(t) = u(t_* + kh) = u_{kh}, \quad t \in [t_* + kh, t_* + (k+1)h], \quad k = \overline{0, N-1}$, $t_* < t^* < \infty$, $h = (t^* - t_*)/N$, N – натуральное число.

В классе дискретных управлений рассмотрим линейную задачу оптимального управления

$$\begin{gathered} J(u) = c^T x(t^*) \longrightarrow \max, \quad \dot{x} = A(t)x + B(t)u + q(t), \quad x(t_*) = x_0, \\ Hx(t^*) = g, \quad |u(t)| \leqslant L, \quad t \in T, \\ x \in \mathbb{R}^n, \quad u \in \mathbb{R}, \quad g \in \mathbb{R}^n, \quad \operatorname{rank} H = m \leqslant n, \end{gathered} \tag{1}$$

Каминцева Виктория Сергеевна – студент, Санкт-Петербургский государственный университет; e-mail: st097171@student.spbu.ru, тел.: +7(904)399-49-72

где $A(t)$, $B(t)$, $q(t)$, $t \in T$ – кусочно-непрерывные $(n \times n)$-матричная и n-векторные функции, $L > 0$. Здесь $x = x(t) \in R^n$ – состояние объекта управления в момент времени t, $u = u(t)$ – значение управляющего воздействия в момент времени t.

Управление $u(\cdot)$ является *допустимым*, если $|u(t)| \leqslant L$, $t \in T$. Каждому допустимому управлению $u(\cdot)$ соответствует единственная траектория $x(\cdot)$ динамической модели (1).

Допустимое управление $u(\cdot)$ – *программа*, если $Hx(t^*) = g$.

Программа $u^0(t)$, $t \in T$, называется *оптимальной* (программным решением задачи (1)), если на соответствующей ей траектории $x^0(\cdot)$ функционал $J(u) = c^T x(t^*)$ достигает максимального значения на множестве всех программ.

Предположим, что в каждый текущий момент времени τ, $\tau \in T_h = [t_*, t_* + h, \ldots, t_* - h]$, будет точно известно текущее состояние $x^*(\tau)$ динамического объекта. Тогда задачу (1) преобразуем к следующему виду

$$\begin{gathered} c^T x(t^*) \longrightarrow \max, \quad \dot{x} = A(t)x + B(t)u + q(t), \quad x(\tau) = z, \\ Hx(t^*) = g, \quad |u(t)| \leqslant L, \quad t \in T(\tau) = [\tau, t^*], \end{gathered} \tag{2}$$

зависящему от позиции (τ, z).

Обозначим через $u^0(t|\tau, z)$, $t \in T(\tau)$, оптимальную программу задачи (2) для позиции (τ, z); X_τ – множество состояний z, для которых задача (2) имеет решение.

Функция

$$u^0(\tau, z) = u^0(\tau|\tau, z), \quad z \in X_\tau, \quad \tau \in T_h, \tag{3}$$

называется *оптимальной дискретной обратной связью (по состоянию)* или *позиционным решением задачи* (1).

Замыкаем дифференциальное уравнение из (1) функцией (3), тем самым получая оптимальную систему управления – конечную цель оптимального синтеза:

$$\dot{x} = A(t)x + B(t)u^0(t, x) + q(t), \quad x(t_*) = x_0. \tag{4}$$

2.2. Система с возмущением. Предположим, что функция (3) построена. Она построена по детерминированной модели, но предназначена для управления динамическим объектом, поведение $x^*(\cdot)$ которого в общем случае отличается от поведения $x(\cdot)$ модели.

Другими словами, оптимальное управление построено для изначальной задачи (1), но в реальности оно будет применено к системе, на которую действует ограниченное кусочно-непрерывное возмущение $w(t)$, в результате чего поведение объекта будет описываться следующим образом:

$$\dot{x} = A(t)x + B(t)u^0(t,x) + q(t) + w(t), \quad x(t_*) = x_0. \tag{5}$$

Это возмущение не известно оптимальному регулятору, но в каждый момент $\tau \in T_h$ ему доступно текущее состояние $x^*(\tau)$.

Таким образом, необходимо построить оптимальную программу для перемещения динамического объекта из начального состояния в конечное при ограничениях, заданных задачей (1), а также при наличии неизвестного возмущения.

3. Переход к интервальной задаче линейного программирования. В статье [10] уже был подробно изложен переход от исходной задачи (1) к ИЗЛП, поэтому далее будут приведены только основные формулы. Данный подход будет применен к задаче (1), в которой отсутствуют возмущения, поскольку регулятору они неизвестны.

После преобразований получена задача линейного программирования

$$\begin{cases} C^T U \longrightarrow \max, \\ DU = g_0, \\ |u_i| \leqslant L, \quad i = \overline{1,N}, \end{cases} \tag{6}$$

где $C = (c_h(t_*), \ldots, c_h(t_* + kh), \ldots, c_h(t^* - h))$ – N-мерный вектор, состоящий из компонент следующего вида

$$c_h(t) = \int_t^{t+h} c^T Y(t^*) Y^{-1}(\tau) B(\tau)\, \mathrm{d}\tau, \quad t \in T_h,$$

а $D = (d_h(t_*), \ldots, d_h(t_* + kh), \ldots, d_h(t^* - h))$ – $(m \times N)$-мерная матрица, состоящая из m-мерных векторов

$$d_h(t) = \int_t^{t+h} G(\tau) B(\tau)\, \mathrm{d}\tau, \quad t \in T_h.$$

$G(\tau) = HY(t^*)Y^{-1}(\tau)$ – матрица размерности $(m \times n)$, а

$$g_0 \;=\; g - HY(t^*)[Y^{-1}(t_*)x_0 - \int\limits_{t_*}^{t^*} Y^{-1}(\tau)q(\tau)\,\mathrm{d}\tau]$$

m-мерный постоянный вектор. $U = (u_0, u_h, \ldots, u_{kh}, \ldots, u_{(N-1)h})^T$ – N-мерный вектор, компонентами которого являются управления на соответствующих промежутках времени.

Используем полученную систему (6) для построения оптимального управления для системы с возмущением (5). Однако нужно учитывать, что реальное положение объекта будет отличаться от траектории, соответствующей вычисленному в начальный момент времени оптимальному управлению $u^0(t_*, x_0)$.

В связи с этим можно рассмотреть циклический процесс: управление начинается в момент t_* с подачи на вход объекта управляющего воздействия $u^*(t) = u^0(t_*, x_0)$, $t \geqslant t_*$. Найденным управлением замыкаем систему с возмущением, и в момент $t = t_* + h$ становится известным состояние объекта $x^*(t_* + h)$, которое в общем случае отлично от $x(t_* + h)$. По этой информации находится новое значение управляющего воздействия $u^0(t_* + h, x^*(t_* + h))$ и т. д.

В каждый момент $\tau \in T_h$ состояние объекта становится известно, и мы можем рассматривать эту точку в качестве начальной для следующего шага построения оптимальной программы на оставшемся промежутке времени. Таким образом происходит коррекция оптимальной программы с целью погасить возникающее возмущение.

4. Численная реализация. Управление электродвигателем. В качестве примера в статье [10] рассматривалась задача управления электродвигателем, модель которого описана в [11], а именно поворот вала двигателя на заданный угол за время T при минимальном расходе энергии.

Уравнение двигателя постоянного тока имеет вид

$$I\ddot{\varphi} = i_{\text{я}} k_\Phi \Phi - M_c, \tag{7}$$

где I – момент инерции вращающейся части двигателя, φ – угол поворота вала двигателя, $i_{\text{я}}$ – ток в якорной цепи, k_Φ – конструктивная постоянная, Φ – магнитный поток, M_c – момент сопротивления.

4.1. Постановка задачи. Используя обозначения

$$x_1 = \varphi, \quad x_2 = \dot{\varphi}, \quad u = i_{\text{я}}, \quad q_c = M_c/I, \quad b = k_\Phi \Phi / I,$$

уравнение (7) можно записать в векторной форме

$$\dot{x} = Ax + Bu + q.$$

В качестве критерия оптимальности рассмотрен функционал

$$\int_{t_*}^{t^*} u(t)\,\mathrm{d}t \longrightarrow \min,$$

где t_* и t^* фиксированы, $t^* - t_* = T$. Краевые условия

$$x(t_*) = x^0, \quad x_1(t^*) = x_1^*, \quad x_2(t^*) = 0.$$

Данная задача сведена к виду (1), при этом многообразие $Hx(t^*) = g$ имеет форму

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_1(t^*) \\ x_2(t^*) \end{pmatrix} = \begin{pmatrix} x_1(t^*) \\ 0 \end{pmatrix}.$$

Итоговая задача линейного программирования (6) примет вид

$$\begin{cases} C^T U \longrightarrow \min, \\ DU = g_0. \end{cases}$$

При наличии возмущения $w(t)$ система будет выглядеть так:

$$\dot{x} = Ax + Bu + q + W(t),$$

$$x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix},\ A = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix},\ B = \begin{pmatrix} 0 \\ b \end{pmatrix},\ q = \begin{pmatrix} 0 \\ -u_c \end{pmatrix},\ W(t) = \begin{pmatrix} 0 \\ w(t) \end{pmatrix}.$$

4.2. Численный эксперимент. Для реализации и тестирования алгоритма были выбраны следующие параметры:

$$\Phi = 0{,}07\text{Вб}; \quad k_\Phi = 28{,}66; \quad I = 2\text{кг} \cdot \text{м}^2; \quad M_c = 2\text{Н} \cdot \text{м}.$$

Начальное положение $x^0 = \begin{pmatrix} 4 \\ 2 \end{pmatrix}$, конечное положение $x(t^*) = \begin{pmatrix} 6 \\ 0 \end{pmatrix}$. Задача управления рассмотрена на промежутке $[t_*, t^*] = [0, 10]$. Этот интервал разбит на $N = 10$ равных временных отрезков, на каждом из которых реализовано кусочно-постоянное управление.

Будем считать, что реализующееся возмущение имеет вид

$$w(t) = \begin{cases} 0, & t \in [0,5], \\ \sin(0{,}1t), & t \in [5,10]. \end{cases}$$

Результаты применения алгоритма показаны на графике (см. рис. 1). Здесь изображена фазовая плоскость переходных процессов в систе-

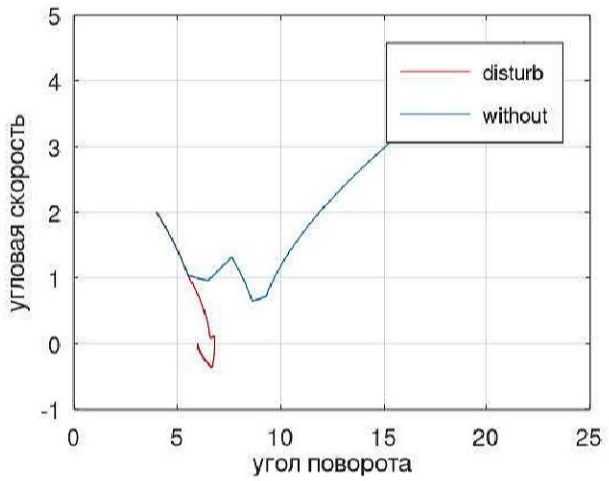


Рис. 1. Фазовая плоскость

ме с возмущением, рассмотренная в двух случаях. В первом случае оптимальная программа была посчитана с учетом возникающих отклонений от начальной траектории (синяя линия). Во втором случае было использовано управление, найденное на первом шаге, т. е. корректировок управления в процессе движения объекта не происходило. Можно заметить, что при отсутствии корректировок управления объект в конечный момент времени находится не в том положении, которого требуют условия задачи. Таким образом, своевременная обратная связь помогает строить управление в режиме реального времени.

5. Заключение. Рассмотренный в данной статье подход может применяться для широкого класса задач, при этом изменение параметров внутри самой задачи (например, число точек переключения

управления, начальнх данных и т.д.) не потребует каких-либо изменений в алгоритме и программной реализации, что является его явным преимуществом.

Первоначальный алгоритм построения оптимального управления хорошо себя показал на линейных системах с линейным функционалом, однако при наличии каких-либо возмущений его применить уже нельзя, поэтому была реализована модификация, основанная на обратной связи. Этот метод имеет высокий потенциал для решения нелинейных задач [12].

Некоторым недостатком данного подхода является необходимость пересчета большого количества значений управлений на каждом шаге. Этого можно избежать, если использовать метод оптимальной опоры [13]. В дальнейшем планируется исследовать проблему построения оптимального управления для нелинейных систем.

Литература

1. Балашевич Н. В., Габасов Р., Кириллова Ф. М. Численные методы программной и позиционной оптимизации линейных систем управления // Журнал вычислительной математики и математической физики. 2000. Т. 40. № 6. С. 836–859.

2. Смирнов Н. В., Пересада В. П., Гидрюк Д. В., Постнов К. В, Попков А. С. Модель межотраслевого баланса – один из базовых элементов информационного обеспечения работы ситуационных центров регионов // Информатизация и связь. 2019. № 3. С. 20–25.

3. Межотраслевой баланс: анализ динамики и управление макроэкономическими тенденциями: учебное пособие для вузов / Н. В. Смирнов, В. П. Пересада, Т. Е. Смирнова [и др.]; Под редакцией Н. В. Смирнова. Санкт-Петербург : Лань, 2021. 180 с.

4. Смирнов Н. В., Смирнова Т. Е., Смирнова М. А., Смирнов М. Н. Разностные динамические модели межотраслевого баланса // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2023. Т. 19. № 1. С. 51–64.

5. Баранов О. В., Попков А. С., Смирнов Н. В. Оптимальная стабилизация квадрокоптера в режиме реального времени // Устойчивость и процессы управления: Материалы III международной

конференции. СПб.: Издательский Дом Федоровой Г.В., 2015. С. 115–116.

6. Baranov O. V., Smirnov N. V., Smirnova T. E., Zholobov Y. V. Design of a quadrocopter with PID-controlled fail-safe algorithm // Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications. 2020. Vol. 11. No 2. P. 23–33.

7. Смирнов Н. В., Соловьева И. В. Применение метода позиционной оптимизации для многопрограммной стабилизации билинейных систем // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2009. № 3. С. 251–259.

8. Смирнов Н. В., Смирнова Т. Е. Синтез многопрограммных управлений в билинейных системах // Прикладная математика и механика. 2000. Т. 64. № 6. С. 929–932.

9. Смирнова М. А., Смирнов М. Н., Смирнов Н. В. Система многоцелевого управления роботом-манипулятором // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2022. Т. 18. № 4. С. 621–630.

10. Каминцева В. С. Реализация метода синтеза оптимального управления для динамических объектов // Процессы управления и устойчивость. 2024. Т. 11. № 1. С. 56–62.

11. Теория автоматического управления. Ч. 2. Теория нелинейных и специальных систем автоматического управления. Изд. 2-е / под ред. А. А. Воронова. М.: Высшая школа, 1986. 504 с.

12. Балашевич Н. В., Габасов Р., Калинин А. И., Кириллова Ф. М. Оптимальное управление нелинейными системами // Журнал вычислительной математики и математической физики. 2002. Т. 42. № 7. С. 969–995.

13. Габасов Р., Кириллова Ф. М. О проблеме синтеза оптимальных систем // Известия Иркутского государственного университета. Серия Математика. 2015. Т. 14. С. 55–63.

УДК 512.647.2

Костеров М. А.

Вычисление коэффициентов параболических аппроксимационно-оценочных критериев при помощи QR-разложения

Рекомендовано к публикации профессором Котиной Е. Д.

1. Введение. Пусть $\mathcal{T} = \{1, \ldots, m\}$, семейство $\xi = \{\xi_t, t \in \mathcal{T}\}$ случайных величин $\xi_t = \xi_t(\omega)$, заданных для любого $t \in \mathcal{T}$ на вероятностном пространстве $(\Omega, \mathcal{F}, \mathsf{P})$, называют дискретным случайным процессом [1], который можно рассматривать как функцию двух переменных $\xi_t(\omega) = \xi(t, \omega)$, где $t \in \mathcal{T}$, $\omega \in \Omega$. Если зафиксировать $t \in \mathcal{T}$, получим случайную величину ξ_t. Если зафиксировать $\omega_0 \in \Omega$, получим последовательность $y_t = \xi_t(\omega_0)$, которую называют траекторией случайного процесса ξ. Далее будем рассматривать только процессы с монотонно возрастающими траекториями.

Задача, которую решают аппроксимационно-оценочные критерии (АОК), заключается в определении момента t_0, в котором траектория y_t переходит от линейного роста к нелинейному [2]. Причем для определения этого момента используются только k значений траектории $y_{t_0-k+1}, y_{t_0-k+2}, \ldots, y_{t_0}$, из левой полуокрестности точки t_0. Эти критерии можно рассматривать как аналитическую формулировку эвристики «метода локтя» [3]. Определим величины

$$\delta_f^2(k_0) = \min_{f \in \mathcal{Y}} \sum_{i=1}^{k} (f(i) - y_{k_0-k+i})^2, \quad \delta_l^2(k_0) = \min_{a,b} \sum_{i=1}^{k} (ai + b - y_{k_0-k+i})^2,$$

где $\mathcal{Y}$ – класс функций вида $a\varphi(x) + b$, φ – непрерывная монотонно возрастающая нелинейная функция и $y_{k_0-k+1}, y_{k_0-k+2}, \ldots, y_{k_0}$ – скользящее окно размера k. Минимумы существуют, поскольку рассматриваются положительно квазиопределенные квадратичные формы [2].

Квадратичные формы $\delta_f^2(k_0)$ и $\delta_l^2(k_0)$ характеризуют «близость» траектории y_t в рассматриваемом окне к функциям из класса $\mathcal{Y}$ и к

Костеров Максим Андреевич – студент, Санкт-Петербургский государственный университет; e-mail: maxim.kosterov@gmail.com, тел.: +7(951)668-77-69

линейным функциям вида $ax + b$ соответственно. Определим теперь величину $\delta^2(k_0) = \delta_l^2(k_0) - \delta_f^2(k_0)$ и значение $t_0 = \min\{k_0 \mid \delta^2(k_0) > 0\}$. Согласно АОК, процесс переходит от линейного роста к нелинейному между точками $t_0 - 1$ и t_0, поскольку окно $y_{t_0-k}, \ldots, y_{t_0-1}$ «ближе» к линейной функции, а окно $y_{t_0-k+1}, \ldots, y_{t_0}$ – уже к нелинейной из класса $\mathcal{Y}$.

В данной работе рассмотрим параболические АОК, т. е. выберем $\varphi(x) = (x-1)^2$. Эти критерии могут быть использованы в механике твердого тела [4, 5], в многомерных статистических методах [6], для нахождения особых точек кинетической кривой ПЦР [7] и автоматического обнаружения аномалий сетевого трафика при DDoS-атаках методами последовательного статистического анализа [8].

2. Квадратичные формы АОК и QR-разложение. Приведем величину $\delta^2(k_0)$ к более удобному виду. Зафиксируем размер окна k и функцию φ. Введем $e = (1, \ldots, 1)^T \in \mathbb{R}^k$ и определим функцию $\rho^2 : \mathbb{R}^k \times \mathbb{R}^k \to \mathbb{R}$,

$$\rho^2(g, x) = \min_{\alpha,\beta}(\alpha g + \beta e - x)^2 = \min_{\alpha,\beta} \sum_{i=1}^{k} (\alpha g_i + \beta - x_i)^2.$$

Зададим векторы $l, f \in \mathbb{R}^k$ через их компоненты: $l_i = i$, $f_i = \varphi(i)$. Обозначим $z = (y_{k_0-k+1}, \ldots, y_{k_0})^T$. Тогда

$$\delta_l^2(k_0) = \min_{a,b} \sum_{i=1}^{k} (ai + b - y_{k_0-k+i})^2 = \min_{\alpha,\beta}(\alpha l + \beta e - z)^2 = \rho^2(l, z),$$

$$\begin{aligned}\delta_f^2(k_0) &= \min_{g\in\mathcal{Y}} \sum_{i=1}^{k} (g(i) - y_{k_0-k+i})^2 = \min_{a,b} \sum_{i=1}^{k} (a\varphi(i) + b - y_{k_0-k+i})^2 = \\ &= \min_{\alpha,\beta}(\alpha f + \beta e - z)^2 = \rho^2(f, z).\end{aligned}$$

Положим, наконец, что $\rho_2 : \mathbb{R}^k \to \mathbb{R}$, $\rho_2(x) = \rho^2(l, x) - \rho^2(f, x)$ и получим

$$\delta^2(k_0) = \delta_l^2(k_0) - \delta_f^2(k_0) = \rho^2(l, z) - \rho^2(f, z) = \rho_2(z). \qquad (1)$$

Таким образом, определение значения величины $\delta^2(k_0)$ сводится к вычислению $\rho_2(x)$ по $x \in \mathbb{R}^k$.

Изучим $\rho^2(g,x)$ как функцию от x при фиксированном g. Пусть $\{g,e\}$ линейно независимы. Введем матрицы $M = (g,e) \in \mathbb{R}^{k\times 2}$ и $S = E - M(M^TM)^{-1}M^T$, где E – единичная матрица. Согласно [9], S – матрица ортогонального проектирования на линейное подпространство, ортогональное $\{g,e\}$, $S^2 = S = S^T$ и

$$\begin{aligned}\rho^2(g,x) &= \min_{\alpha,\beta}(\alpha g + \beta e - x)^2 = (Sx)^2 = x^TS^TSx = x^TSx = \\ &= x^2 - x^TM(M^TM)^{-1}M^Tx.\end{aligned} \tag{2}$$

Для решения задач линейного метода наименьших квадратов используют QR-разложение [10], поэтому рассмотрим

$$\widetilde{g} = g - \frac{e^Tg}{e^2}e, \quad Q = (\widetilde{g},e), \quad L = \begin{pmatrix} 1 & 0 \\ e^Tg/e^2 & 1 \end{pmatrix}.$$

Заметим, что $\widetilde{g}^Te = 0$ и L невырождена. Нетрудно убедиться, что $M = QL$. Тогда

$$\begin{aligned}M(M^TM)^{-1}M^T &= QL(L^T(Q^TQ)L)^{-1}L^TQ^T = \\ &= QLL^{-1}(Q^TQ)^{-1}L^{-T}L^TQ^T = Q(Q^TQ)^{-1}Q^T,\end{aligned}$$

$$(Q^TQ)^{-1} = \begin{pmatrix} \widetilde{g}^2 & \widetilde{g}^Te \\ \widetilde{g}^Te & e^2 \end{pmatrix}^{-1} = \begin{pmatrix} \widetilde{g}^2 & 0 \\ 0 & e^2 \end{pmatrix}^{-1} = \begin{pmatrix} 1/\widetilde{g}^2 & 0 \\ 0 & 1/e^2 \end{pmatrix}.$$

Поскольку $\{g,e\}$ линейно независимы, то $\widetilde{g} \neq 0$ и $(Q^TQ)^{-1}$ определена. Таким образом, по формуле (2) получим

$$\begin{aligned}\rho^2(g,x) &= x^2 - x^TQ(Q^TQ)^{-1}Q^Tx = \\ &= x^2 - \begin{pmatrix} \widetilde{g}^Tx & e^Tx \end{pmatrix}\begin{pmatrix} 1/\widetilde{g}^2 & 0 \\ 0 & 1/e^2 \end{pmatrix}\begin{pmatrix} \widetilde{g}^Tx \\ e^Tx \end{pmatrix} = \\ &= x^2 - \frac{(\widetilde{g}^Tx)^2}{\widetilde{g}^2} - \frac{(e^Tx)^2}{e^2},\end{aligned}$$

$$\begin{aligned}\rho_2(x) = \rho^2(l,x) - \rho^2(f,x) &= \left(x^2 - \frac{(e^Tx)^2}{e^2} - \frac{(\widetilde{l}^Tx)^2}{\widetilde{l}^2}\right) - \\ - \left(x^2 - \frac{(e^Tx)^2}{e^2} - \frac{(\widetilde{f}^Tx)^2}{\widetilde{f}^2}\right) &= \frac{(\widetilde{f}^Tx)^2}{\widetilde{f}^2} - \frac{(\widetilde{l}^Tx)^2}{\widetilde{l}^2},\end{aligned} \tag{3}$$

где $\widetilde{f}$ и $\widetilde{l}$ определены по аналогии с $\widetilde{g}$. Нетрудно убедиться, что предположение о линейной независимости $\{f, e\}$ и $\{l, e\}$ выполняется.

3. Вычислительная схема. Воспользуемся формулой (3) для составления вычислительной схемы для параболического АОК. Рассмотрим $g \in \mathbb{R}^k$,

$$\widetilde{g}^2 = \left(g - \frac{e^T g}{e^2} e\right)^2 = g^2 + \frac{(e^T g)^2}{e^2} - 2\frac{(e^T g)^2}{e^2} = g^2 - \frac{(e^T g)^2}{e^2}.$$

Введем

$$g^\Sigma = \frac{e^T g}{e^2} = \frac{1}{k}\sum_{i=1}^{k} g_i, \quad g^V = \widetilde{g}^2 = \sum_{i=1}^{k} g_i^2 - k(g^\Sigma)^2.$$

Заметим, что тогда $\widetilde{g}_i = g_i - g^\Sigma$. По формулам из [11] определим многочлены

$$F_1(n) = \frac{1}{2}n^2 + \frac{1}{2}n = \sum_{i=1}^{n} i, \quad F_2(n) = \frac{1}{3}n^3 + \frac{1}{2}n^2 + \frac{1}{6}n = \sum_{i=1}^{n} i^2,$$

$$F_4(n) = \frac{1}{5}n^5 + \frac{1}{2}n^4 + \frac{1}{3}n^3 - \frac{1}{30}n = \sum_{i=1}^{n} i^4.$$

Определим по аналогии с g^Σ и g^V значения

$$l^\Sigma = \frac{1}{k}\sum_{i=1}^{k} i = \frac{1}{k}F_1(k), \tag{4}$$

$$l^V = \sum_{i=1}^{k} i^2 - k(l^\Sigma)^2 = F_2(k) - \frac{F_1^2(k)}{k}, \tag{5}$$

$$f^\Sigma = \frac{1}{k}\sum_{i=1}^{k} (i-1)^2 = \frac{1}{k}F_2(k-1), \tag{6}$$

$$f^V = \sum_{i=1}^{k} (i-1)^4 - k(f^\Sigma)^2 = F_4(k-1) - \frac{F_2^2(k-1)}{k}. \tag{7}$$

По формуле (3) получим

$$\rho_2(x) = \frac{1}{f^V}\left(\sum_{i=1}^{k}((i-1)^2 - f^\Sigma)x_i\right)^2 - \frac{1}{l^V}\left(\sum_{i=1}^{k}(i - l^\Sigma)x_i\right)^2.$$

Тогда, согласно (1),

$$\begin{aligned}\delta^2(k_0) =& \frac{1}{f^V}\left(\sum_{i=1}^{k}((i-1)^2 - f^\Sigma)y_{k_0-k+i}\right)^2 - \\ &- \frac{1}{l^V}\left(\sum_{i=1}^{k}(i - l^\Sigma)y_{k_0-k+i}\right)^2.\end{aligned} \tag{8}$$

Вычислительная схема состоит из двух этапов: вычисление вспомогательных значений l^Σ, l^V, f^Σ, f^V по формулам (4)–(7) и вычисление $\delta^2(k_0)$ по формуле (8).

Например, для $k = 5$

$$l^\Sigma = 3, \quad l^V = 10, \quad f^\Sigma = 6, \quad f^V = 174,$$

$$\begin{aligned}\delta^2(k_0) &= \frac{1}{174}\left(\sum_{i=1}^{5}((i-1)^2 - 6)y_{k_0-k+i}\right)^2 - \\ &- \frac{1}{10}\left(\sum_{i=1}^{5}(i-3)y_{k_0-k+i}\right)^2 = \\ &= \frac{1}{174}(-6y_{k_0-4} - 5y_{k_0-3} - 2y_{k_0-2} + 3y_{k_0-1} + 10y_{k_0})^2 - \\ &- \frac{1}{10}(-2y_{k_0-4} - y_{k_0-3} + y_{k_0-1} + 2y_{k_0})^2.\end{aligned} \tag{9}$$

Зафиксировав $y_{k_0-4} = 0$, раскрыв скобки, получим

$$\begin{aligned}\delta^2(k_0) = \frac{1}{435}(&19y_{k_0-3}^2 + 10y_{k_0-2}^2 - 21y_{k_0-1}^2 + 76y_{k_0}^2 - 24y_{k_0-1}y_{k_0} + \\ &+ y_{k_0-3}(50y_{k_0-2} + 12y_{k_0-1} - 76y_{k_0}) + y_{k_0-2}(-30y_{k_0-1} - 100y_{k_0})),\end{aligned} \tag{10}$$

что после переобозначения y_{k_0-5+i} как y_i совпадает с формулой, полученной в работе [2]. На практике рекомендуется не раскрывать

скобки, а пользоваться более простой формулой (9) или, в общем случае, формулой (8).

4. Заключение. В работе [2] предложен алгоритм вычисления $\delta^2(k_0)$, в котором используется метод скользящего окна и преобразование $\widetilde{y}_i = y_{k_0-k+1+i} - y_{k_0-k+1}$. Заметим, что δ^2 совпадает для окон $\widetilde{y}_0, \ldots, \widetilde{y}_{k-1}$ и $y_{k_0-k+1}, \ldots, y_{k_0}$. Значения $\widetilde{y}_1, \ldots, \widetilde{y}_{k-1}$ (известно, что $\widetilde{y}_0 = 0$) подставляют в заранее выведенные формулы, аналогичные (10). Эти формулы получают отдельно для каждого k.

Сравним этот алгоритм с построенной вычислительной схемой. Предложенный ранее алгоритм требует хранения вычисленных коэффициентов, количество которых растет с ростом k, а новая вычислительная схема при любом k использует только четыре значения: l^{Σ}, l^{V}, f^{Σ}, f^{V}. Сложность вычисления коэффициентов разложения растет с ростом k. Получение четырех чисел схемы сводится к нахождению значений многочленов по степеням k, причем число операций не зависит от k. Алгоритм в [2] подразумевает реализацию отдельной формулы для каждого используемого k, а новая вычислительная схема позволяет реализовать процесс вычисления величины $\delta^2(k_0)$ для произвольного k и открывает возможность применения АОК для больших k.

Литература

1. Булинский А. В., Ширяев А. Н. Теория случайных процессов. М.: Физматлит, 2004. 401 с.

2. Orekhov A. V. Quasi-deterministic processes with monotonic trajectories and unsupervised machine learning // Mathematics. 2021. Vol. 9. No 18. Art. no 2301.

3. Thorndike R. L. Who belongs in the family? // Psychometrika. 1953. Vol. 18. P. 267–276.

4. Orekhov A. V. Statistical criteria for the limits of application of Hooke's law // Vestnik of Saint Petersburg University. Applied Mathematics. Computer Science. Control Processes. 2020. Vol. 16. No 4. P. 391–401.

5. Орехов А. В. Машинное обучение и определение момента перехода к следующей стадии ползучести // X Поляховские чтения: Материалы международной научной конференции по механике. СПб.: ООО «Издательство ВВМ», 2024. С. 713–717.

6. Орехов А. В., Васильев Е. В. Бинарные метрические деревья и иерархия вложенных кластеров // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2024. Т. 20. № 4. С. 487–499.

7. Orekhov A. V., Potekhina M. A. Unsupervised machine learning methods for determining special points of the polymerase chain reaction fluorescence accumulation curve // Moscow University Physics Bulletin. 2023. Vol. 78. No 1. P. S169–S179.

8. Орехов А. В., Орехов А. А. Автоматическое обнаружение аномалий сетевого трафика при DDoS-атаках // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2023. Т. 19. № 2. С. 251–263.

9. Буре В. М., Парилина Е. М. Теория вероятностей и математическая статистика. СПб.: Лань, 2013. 416 с.

10. Nocedal J., Wright S. J. Numerical Optimization. New York: Springer, 2006. 664 p.

11. Двайт Г. Б. Таблицы интегралов и другие математические формулы. Изд. 5-е. М.: Наука, 1978. 224 с.

УДК 519.6

Семенова А. Ю.

Многокритериальная оптимизация параметров управления динамикой пучка заряженных частиц в линейном ускорителе

Рекомендовано к публикации доцентом Владимировой Л. В.

1. Введение. В настоящей работе рассматривается линейный волноводный ускоритель электронов, широко применяемый в медицинских целях. Процессы в этом ускорителе исследуются многими авторами [1–6], однако задача оптимизации динамики пучка остается актуальной и в настоящее время в связи с разнообразием устройств и областей их применения.

Для описания продольной динамики пучка используется подход, предложенный А. Д. Овсянниковым [7]. Эволюция пучка описывается как совокупность программного движения (динамики синхронной частицы) и семейства движений частиц пучка. Управляющими функциями являются безразмерный параметр амплитуды напряженности поля ускоряющей волны и закон изменения синхронной фазы.

Рассматриваются параметризованные управляющие функции, полученные в [8], с некоторым начальным вектором параметров. Оптимизация динамики пучка заряженных частиц проводится по параметрам управляющих функций с использованием нескольких критериев качества.

2. Постановка задачи. Основной задачей в работе с линейными ускорителями является выбор электромагнитных полей, обеспечивающих необходимые характеристики пучка на выходе ускорителя. Применительно к продольному движению было выбрано три критерия качества для оценки движения частиц: $K_1(u), K_2(u), K_3(u)$ – захват частиц в режим ускорения, фазовая и энергетическая неоднородность пучка соответственно. Вектор параметров $u \in D \subset E^r$, определяющий управления, принимает значения в ограниченной области D.

Семенова Анастасия Юрьевна – магистрант, Санкт-Петербургский государственный университет; e-mail: a2011se@gmail.com, тел.: +7(911)140-77-48

В результате необходимо решать задачу многокритериальной оптимизации:

$$\min_{u \in D} K(u)$$

по вектору параметров u, определяющему управления, здесь $K(u) = (K_1(u), K_2(u), K_3(u))$ – вектор критериев.

Такой подход оказывается эффективнее в ситуации с несколькими противоречивыми критериями и позволяет достигнуть сразу несколько оптимизационных целей.

Для решения этой задачи необходимо: в контексте модели продольной динамики пучка смоделировать новые управления, исходя из возможностей по варьированию параметров, рассчитать значения критериев качества для полученных управлений и выделить из вектор-критериев Парето-оптимальное множество. Множество управлений, соответствующих Парето-оптимальным точкам, будет являться множеством эффективных управлений.

3. Модель продольной динамики пучка. Пусть динамический управляемый процесс описывается системой дифференциальных уравнений [7]:

$$\begin{cases} \dfrac{\mathrm{d}\gamma_s}{\mathrm{d}\xi} = -\alpha(\xi, u)\sin\left(\phi_s(\xi, u)\right), \\ \dfrac{\mathrm{d}\gamma}{\mathrm{d}\xi} = -\alpha(\xi, u)\sin\left(\widehat{\phi} + \phi_s(\xi, u)\right), \\ \dfrac{\mathrm{d}\hat{\phi}}{\mathrm{d}\xi} = 2\pi\left(\dfrac{\gamma_s}{\sqrt{\gamma_s^2 - 1}} - \dfrac{\gamma}{\sqrt{\gamma^2 - 1}}\right), \end{cases} \tag{1}$$

с начальными условиями для N частиц в пучке: $\gamma(0) = \gamma_0$, $\gamma_s(0) = \gamma_s 0$, $\widehat{\phi} = \widehat{\phi}_{0n}, \widehat{\phi}_{0n} = -\phi_{s0} + \frac{\pi}{N} + \frac{2\pi}{N}(n-1), n = \overline{1, N}$. Частицы по фазам заполняют промежуток $(-\pi, \pi)$.

Первое уравнение системы (1) описывает динамику синхронной частицы, второе – динамику частиц пучка, а третье уравнение – изменение фазы пучка в отклонении от синхронной фазы. Здесь ξ, γ – приведенная продольная координата и приведенная энергия частицы соответственно; $\xi \in [0, L]$, L – приведенная длина ускорителя. Характеристики синхронной частицы снабжены индексом s. Переменная $\widehat{\phi} = \phi - \phi_s$ характеризует отклонение фазы частицы от синхронной фазы ϕ_s.

Управления в (1) задаются двумя функциями: $\alpha(\xi, u)$ – безразмерный параметр амплитуды поля ускоряющей волны, а $\phi_s(\xi, u)$ – закон изменения синхронной фазы вдоль структуры.

3.1. Параметрические управления. Параметрические управления, найденные в [8], имеют вид

$$\begin{aligned}\alpha(\xi, u) = u_1 \left(1 - \frac{1}{1 + u_2|\xi - u_3|}\right) \operatorname{sign}(\xi - u_3) + u_1 + \\ + u_4 \left(1 - \frac{1}{1 + u_5|\xi - u_6|}\right) \operatorname{sign}(\xi - u_6) + u_4 + \\ + u_7 \left(1 - \frac{1}{1 + u_8|\xi - u_9|}\right) \operatorname{sign}(\xi - u_9) + u_7,\end{aligned} \tag{2}$$

$$\begin{aligned}\phi_s(\xi, u) = -u_{10} \left(1 - \frac{1}{1 + u_{11}|\xi - u_{12}|}\right) \operatorname{sign}(\xi - u_{12}) - \\ - u_{10} + \frac{u_{10}}{1 + u_{11}u_{12}}.\end{aligned} \tag{3}$$

Обозначим параметры управлений из формул (2), (3) вектором $u = (u_1, \ldots, u_{12})$. Начальное значение вектора параметров $u = u^{(0)}$, полученное в [8], обеспечивает приемлемую динамику пучка. Опираясь на полученный вектор $u^{(0)}$, можно предложить диапазон изменения каждой координаты вектора параметров с учетом его математического смысла и, тем самым, определить область $D \subset E^{12}$.

4. Критерии оценки качества пучка и цели оптимизации. Приведенные критерии качества являются интегральными по сечению ансамбля траекторий рассматриваемого динамического процесса и содержат фазовую плотность в подынтегральной функции. Такие интегралы допускают вероятностную интерпретацию и вычисляются по методу Монте-Карло [9]. Целями оптимизации являются:

1. Минимизация критерия захвата частиц в режим ускорения:

$$\begin{aligned}K_1(u) = \int_0^L \int_{M_{\xi,u}} \Phi(\phi_\xi)\rho(\xi, \gamma_\xi, \phi_\xi)\, d\gamma_\xi\, d\phi_\xi\, d\xi \approx \\ \approx \frac{L}{2SN} \left(\sum_{k=1}^{N} \Phi(\phi_0^{(k)}) + \sum_{k=1}^{N} \Phi(\phi_L^{(k)}) + 2 \sum_{i=1}^{S} \sum_{k=1}^{N} \Phi(\phi_{\xi_i}^{(k)}) \right),\end{aligned} \tag{4}$$

где S – число сечений, в которых наблюдаются фаза пучка; N – число модельных частиц; L – приведенная длина ускорителя; $M_{\xi,u}$ – сечение траекторий пучка в момент значения продольной координаты ξ при управлениях с выбранным вектором параметров u; $\rho(\xi, \gamma_\xi, \phi_\xi)$ – плотность распределения фазового вектора (γ_ξ, ϕ_ξ), где ϕ_ξ, γ_ξ – фаза частиц и энергия частиц при значении продольной координаты ξ. Функция, входящая в (4),

$$\Phi(\phi) = \begin{cases} (\phi - \pi)^2, & \phi \leqslant \pi, \\ 0, & \phi \in (-\pi, \pi), \\ (\phi + \pi)^2, & \phi \geqslant \pi, \end{cases}$$

является штрафной функцией, которая позволяет определить попадание переменной ϕ в заданный интервал $(-\pi, \pi)$.

2. Минимизация фазовой неоднородности пучка на выходе ускорителя:

$$\begin{aligned} K_2(u) &= \int_{M_L,u} (\phi_L - \overline{\phi}_L)^2 \rho(L, \gamma_L, \phi_L)\, d\gamma_L\, d\phi_L \approx \\ &\approx \frac{1}{N_L} \sum_{k=1}^{N_L} \left(\phi_L^{(k)} - \overline{\phi}_L)\right)^2, \end{aligned} \tag{5}$$

$$\overline{\phi}_L = \int_{M_L,u} \phi_L \rho(L, \gamma_L, \phi_L)\, d\gamma_L\, d\phi_L \approx \frac{1}{N_c} \sum_{k=1}^{N_c} \phi_L^{(k)}.$$

Здесь N_L – число частиц, достигших выхода структуры (индекс L показывает, что рассматриваются характеристики на выходе ускорителя); N_c – число частиц, захваченных в режим ускорения, т. е. тех, фазы которых в ходе движения не вышли из промежутка $(-\pi, \pi), (N_c \leqslant N_L)$.

3. Минимизация энергетической неоднородности пучка на выходе ускорителя:

$$\begin{aligned} K_3 &= \int_{M_L,u} (\gamma_L - \widehat{\gamma})^2 \rho(L, \gamma_L, \phi_L)\, d\gamma_L\, d\phi_L \approx \\ &\approx \frac{1}{N_L} \sum_{k=1}^{N_L} \left(\gamma_L^{(k)} - \widehat{\gamma}\right)^2. \end{aligned} \tag{6}$$

Здесь N_L – число частиц, достигших выхода структуры, $\widehat{\gamma}$ – заданная энергия на выходе ускорителя, которую необходимо достичь.

5. Многокритериальная оптимизация. Решение оптимизационной задачи для модели (1) находится как множество Парето в пространстве критериев $K_1(u), K_2(u), K_3(u)$ при условии $K_1 = 0$. В этом случае все частицы достигают конца ускорителя и не выходят за промежуток $(-\pi, \pi)$ при управлении, определяемом вектором параметров u.

Алгоритм многокритериальной оптимизации:

1. Будем выбирать вектор параметров u как случайный равномерно распределенный вектор в области $D \subset E^{12}$:

$$\begin{aligned} D = ([0{,}005; 0{,}1] \times [0{,}1; 25{,}0] \times [1{,}0; 1{,}5] \times [0{,}45; 0{,}7] \times \\ \times [3{,}0; 6{,}0] \times [2{,}2; 2{,}7] \times [0{,}5; 0{,}8] \times [1{,}0; 5{,}0] \times \\ \times [3{,}2; 3{,}8] \times [0{,}4; 1{,}2] \times [0{,}4; 4{,}5] \times [2{,}5; 3{,}7]). \end{aligned} \tag{7}$$

2. Пусть $u^{(k)} \in D$ выбранный вектор, через который определяются управления $\alpha(\xi, u^{(k)})$ и $\phi_s(\xi, u^{(k)})$ по формулам (2), (3).
3. После численного интегрирования системы (1) по формулам (4)–(6) вычисляем значения для $K_1(u^{(k)}), K_2(u^{(k)}), K_3(u^{(k)})$.
4. Делаем проверку $K_1(u^{(k)}) = 0$.
5. Если условие 4 выполнено, вектор параметров $u^{(k)}$ является допустимым. В противном случае этот вектор отбрасывается.

Процедура алгоритма 1–5 выполняется до получения M допустимых векторов $u^{(k)}$ из области варьирования параметров D. В пространстве критериев K_2, K_3 получаем M соответствующих точек $(K_2(u^{(k)}), K_3(u^{(k)})$, $k = \overline{1, M}$. Далее применяем алгоритм нахождения компромиссной кривой, описанный в [9].

6. Численные результаты. Задача оптимизации продольной динамики пучка решалась со следующими параметрами:

- начальная энергия $W_0 = 80$ кэВ;
- длина ускорителя $\overline{L} = 0{,}8$ м;
- длина ускоряющей волны $\lambda = 0{,}1$ м;
- требуемая приведенная энергия на выходе ускорителя $\widehat{\gamma} = 12$.

При заданной энергии W_0 начальная приведенная энергия частиц будет иметь значение $\gamma_0 = \gamma_{s0} = 1{,}15655$.

Генерировалось 700 новых вектор-параметров $u^{(k)}, k = \overline{1,700}$, равномерно распределенных в области D, определяемой (7). Они определяют управления $\alpha(\xi, u^{(k)}$ и $\phi_s(\xi, u^{(k)})$. При полученных управлениях система (1) интегрировалась явным методом Рунге – Кутты четвертого порядка при шаге интегрирования $h = 0{,}01$.

Для сгенерированных управляющих векторов были проведены оценки согласно критериям (4)–(6) для выявления M допустимых управлений. Если значение критерия $K_1 > 0$ для некоторого вектора параметров $u^{(k)}$, то точка в пространстве (K_2, K_3), соответствующая этому вектору, отмечалась треугольником (рис. 1). В противном случае точка отмечалась кружком. Для точек, отмеченных кружком, находилось Парето-оптимальное множество – множество неулучшаемых векторов, при которых все частицы достигают конца ускорителя и не выходят за промежуток $(-\pi, \pi)$.

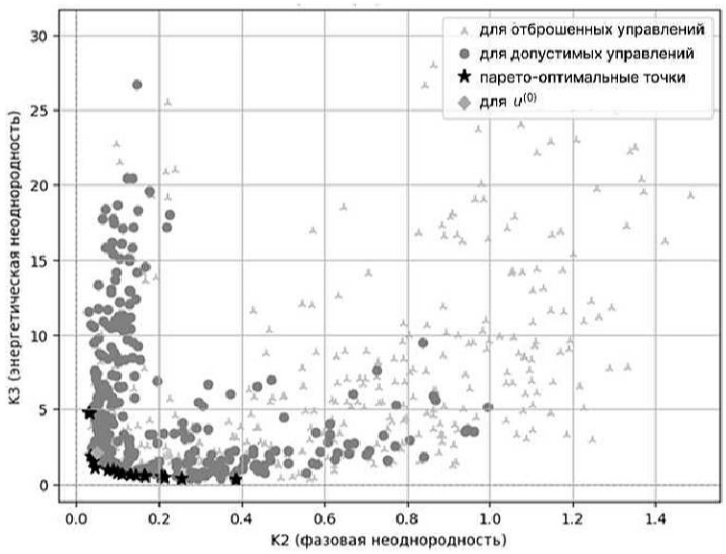


Рис. 1. Множество Парето (отмечено звездочками) в пространстве критериев (K_2, K_3)

Таблица 1. Значения параметров, определяющих эффективные управления, и значения параметров $u^{(0)}$, полученных в [8]

№	$u^{(1)}$	$u^{(2)}$	$u^{(3)}$	$u^{(4)}$	$u^{(5)}$	$u^{(6)}$	$u^{(7)}$	$u^{(8)}$	$u^{(0)}$
u_1	0,066	0,029	0,070	0,051	0,095	0,011	0,020	0,026	0,039
u_2	6,757	10,637	2,995	1,877	2,836	0,627	23,320	9,577	2,497
u_3	1,103	1,013	1,001	1,386	1,165	1,173	1,438	1,076	1,15
u_4	0,599	0,685	0,688	0,684	0,656	0,654	0,691	0,590	0,546
u_5	2,226	2,206	2,319	2,925	1,994	2,478	1,498	2,795	4,497
u_6	2,329	2,221	2,205	2,231	2,636	2,219	2,310	2,614	2,35
u_7	0,610	0,699	0,515	0,562	0,631	0,682	0,628	0,730	0,741
u_8	1,944	1,759	1,967	1,760	1,713	1,701	1,579	1,567	2,351
u_9	3,492	3,503	3,381	3,314	3,523	3,573	3,621	3,788	3,5
u_{10}	0,641	0,640	0,910	0,722	0,867	0,848	0,789	0,709	0,645
u_{11}	1,249	1,596	1,934	2,819	1,244	3,413	2,694	3,468	4,474
u_{12}	3,562	3,470	3,371	3,001	3,660	3,558	3,366	3,206	3,2

Таблица 2. Значения критериев для параметров эффективных управлений и для параметров $u^{(0)}$, полученных в [8]

№	$u^{(1)}$	$u^{(2)}$	$u^{(3)}$	$u^{(4)}$	$u^{(5)}$	$u^{(6)}$	$u^{(7)}$	$u^{(8)}$	$u^{(0)}$
K_1	0	0	0	0	0	0	0	0	8,28
K_2	0,030	0,036	0,385	0,167	0,163	0,255	0,206	0,143	0,049
K_3	4,835	1,890	0,351	0,566	0,588	0,387	0,554	0,636	2,08

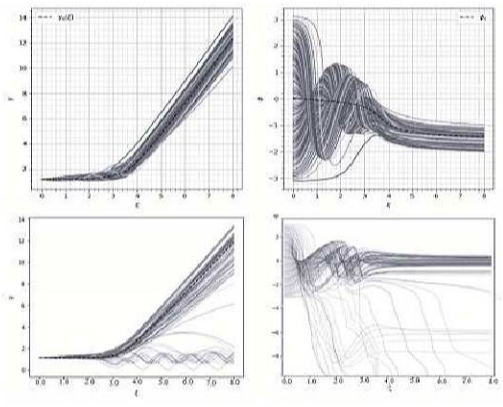


Рис. 2. Динамика пучка при управляющем векторе $u^{(5)}$ (сверху) и векторе $u^{(0)}$ (снизу)

На рис. 2 представлена динамика частиц пучка: слева изменение энергии вдоль ускорителя, справа – изменение фазы. Верхние два рисунка получены с одним из вариантов эффективных управлений $\alpha(\xi, u^{(5)})$, $\phi_s(\xi, u^{(5)})$, а нижние два – с управлениями $\alpha(\xi, u^{(0)})$, $\phi_s(\xi, u^{(0)})$, зависящими от начального вектора параметров. Все значения вектор-параметров представлены в таблице 1.

Рис. 2 и таблица 2 показывают, что начальный вектор параметров $u^{(0)}$ определяет управления, дающие приемлемую динамику, но для полного захвата частиц в режим ускорения потребовалась оптимизация.

7. Выводы. Управляющие векторы параметров, полученные в результате оптимизации, дают эффективные управления, которые обеспечивают полный захват частиц пучка в режим ускорения. В отличие от исходного набора параметров $u^{(0)}$, для которого нарушается условие $K_1(u^{(0)}) = 0$, они позволяют полностью исключить потерю частиц. Кроме того, результаты исследования показывают, что при полном захвате частиц в ускоряющий режим достигнутая на выходе энергия соответствует заданным требованиям.

Так как критерии противоречивы, при достижении минимума фазовой неоднородности энергетическая неоднородность оказывается максимальной, и наоборот.

Однако, множество Парето включает достаточно много точек для выбора в зависимости от преследуемых задач. Это формирует пространство альтернатив, где каждое решение представляет собой эффективную стратегию управления.

Литература

1. Вальднер О. А., Власов А. Д., Шальнов А. В. Линейные ускорители. М.: Атомиздат, 1969. 248 с.

2. Власов А. Д. Теория линейных ускорителей. М.: Атомиздат, 1965. 308 с.

3. Капчинский И. М. Теория линейных резонансных ускорителей. Динамика частиц. М.: Энергоиздат, 1982. 240 с.

4. Овсянников Д. А., Егоров Н. В. Математическое моделирование систем формирования электронных и ионных пучков. СПб.: Изд-во С.-Петерб. ун-та, 1998. 276 с.

5. Овсянников А. Д. Управление пучком заряженных частиц с учетом их взаимодействия // Вестник Санкт-Петербургского университета. Серия 10: Прикладная математика. Информатика. Процессы управления. 2009. № 2. С. 82–92.

6. Vladimirova L. V., Rubtsova I. D., Zhdanova A. Y., Edamenko N. S. Genetic stochastic algorithm application in beam dynamics optimization // Problem Stability and Control Processes – Proceedings of the 4th International Conference Dedicated to the Memory of Professor Vladimir Zubov, Springer International Publishing (Lecture Notes in Control and Information Sciences - Proceedings). 2022. P. 265–272.

7. Ovsyannikov A. D., Shirokolobov A. Y. Mathematical model of beam dynamics optimization in traveling wave // Proceedings of the RuPAC'12. St. Petersburg. 2012. P. 355–357.

8. Овсянников Д. А., Рубцова И. Д., Владимирова Л. В., Едаменко Н. С., Бушин А. Р. Об одном подходе к параметризации управляющих функций в задачах оптимизации // Сборник научных трудов X Международной конференции. М.: Национальный исследовательский ядерный университет МИФИ, 2024. С. 120–121.

9. Владимирова Л. В., Овсянников Д. А., Рубцова И. Д. Методы Монте-Карло в прикладных задачах. СПб.: ВВМ, 2015. 167 с.

УДК 517.95

Соколов В. М.

Функционал Ляпунова – Красовского с заданной производной для волнового уравнения

Рекомендовано к публикации доцентом Егоровым А. В.

1. Введение. Исследование волновых уравнений является важной задачей в математической физике и теории управления, поскольку они описывают широкий класс явлений, включая распространение упругих, акустических и электромагнитных волн. Одним из ключевых аспектов анализа таких уравнений является изучение их устойчивости, что играет критическую роль в понимании динамического поведения решений и разработке надёжных методов управления.

В настоящее время разработан метод построения функционалов с заданной производной для дифференциальных систем с запаздыванием без частных производных [1, 2]. Эти функционалы позволяют строить оценки решений, исследовать системы на устойчивость и робастную устойчивость. В данной работе эти результаты обобщены на случай волнового уравнения. Построен функционал Ляпунова – Красовского с простейшей квадратичной производной, исследована матрица Ляпунова, являющаяся ключевым элементом функционала.

2. Постановка задачи. Рассмотрим волновое дифференциальное уравнение [3, 4]

$$u_{tt}(x,t) = a^2 u_{xx}(x,t) + bu_t(x,t) + cu(x,t), \quad x \in (0,l),\ t > 0, \qquad (1)$$

с граничным условием $u(0,t) = u(l,t) = 0,\ t \geqslant 0$, и начальными условиями

$$u(x,0) = \varphi_0(x), \quad u_t(x,0) = \varphi_1(x), \quad x \in [0,l].$$

Соколов Вадим Михайлович – аспирант, Санкт-Петербургский государственный университет; e-mail: vad_sokolov@mail.ru, тел.: +7(921)335-02-84

Исследование выполнено за счет гранта Российского научного фонда № 23-71-10099, https://rscf.ru/project/23-71-10099/

При этом выполняется

$$\varphi_0(0) = \varphi_0(l) = 0, \;\; \varphi_1(0) = \varphi_1(l) = 0.$$

Не умаляя общности, будем считать, что $l = 1$.

Построим квадратичный непрерывный функционал v, который будет удовлетворять условиям

$$\frac{d}{dt} v(\widehat{u}_t(\varphi)) = -\|\widehat{u}_t(\varphi)\|^2_{L_2}, \quad v(0) = 0, \tag{2}$$

где $\widehat{u}_t(\varphi)$ – состояние системы в момент времени t при начальном состоянии $\varphi(x) = (\varphi_0(x), \varphi_1(x))^T$, $x \in [0, 1]$.

3. Применение метода Фурье. Построим общее решение краевой задачи методом Фурье. Пусть

$$u(x, t) = F(x)G(t).$$

Подставим произведение в уравнение (1). Тогда

$$F(x)G''(t) = a^2 F''(x)G(t) + bF(x)G'(t) + cF(x)G(t).$$

Отсюда:

$$a^2 \frac{F''(x)}{F(x)} = \frac{G''(t) - bG'(t) - cG(t)}{G(t)} = a^2\lambda,$$

где λ – некоторый параметр.

Таким образом, получим систему уравнений

$$\begin{cases} F''(x) = \lambda F(x), \\ G''(t) - bG'(t) - cG(t) = a^2\lambda G(t). \end{cases} \tag{3}$$

Проанализируем все возможные значения параметра: 1) $\lambda = n^2$, 2) $\lambda = 0$, 3) $\lambda = -n^2$. Первое уравнение системы (3) при условии выполнения граничных условий $u(0, t) = u(1, t) = 0$, $t \geqslant 0$, будет иметь решение $F(x) = \sin(nx)$, где $n = \pi i$, $i \in \mathbb{N}$, только при $\lambda = -n^2$. Пронумеруем полученные функции: $F_i(x) = \sin(\pi i x)$, $i \in \mathbb{N}$.

Через G_i обозначим решения второго уравнения системы (3) при $\lambda = -\pi^2 i^2$. Эта функция удовлетворяет системе

$$\begin{cases} G_i'(t) = H_i(t), \\ H_i'(t) = (-a^2\pi^2 i^2 + c)G_i(t) + bH_i(t). \end{cases}$$

В матричном виде система принимает вид

$$\begin{pmatrix} G_i'(t) \\ H_i'(t) \end{pmatrix} = A_i \begin{pmatrix} G_i(t) \\ H_i(t) \end{pmatrix}, \tag{4}$$

где $A_i = \begin{pmatrix} 0 & 1 \\ -a^2\pi^2 i^2 + c & b \end{pmatrix}$.

Применим к полученной системе уравнений (4) формулу Коши:

$$\begin{pmatrix} G_i(t) \\ H_i(t) \end{pmatrix} = K_i(t) \begin{pmatrix} G_{i0} \\ H_{i0} \end{pmatrix},$$

где $K_i = e^{A_i t}$ – фундаментальная матрица системы (4), G_{i0}, H_{i0} – произвольные константы.

Общее решение имеет вид

$$u(x,t) = \sum_{i=1}^{\infty} F_i(x) G_i(t).$$

Разложим начальные функции в ряды Фурье:

$$\varphi_0(x) = \sum_{i=1}^{\infty} \varphi_{0i} \sin(\pi i x), \quad \varphi_1(x) = \sum_{i=1}^{\infty} \varphi_{1i} \sin(\pi i x),$$

где

$$\varphi_{0i} = 2\int_0^1 \varphi_0(x) \sin(\pi i x) dx, \ \varphi_{1i} = 2\int_0^1 \varphi_1(x) \sin(\pi i x) dx.$$

Константы G_{i0}, H_{i0}, $i \in \mathbb{N}$, найдём из начальных условий, опираясь на разложения функций φ_0 и φ_1:

$$u(x,0) = \sum_{i=1}^{\infty} F_i(x) G_{i0} = \sum_{i=1}^{\infty} F_i(x) \varphi_{0i} = \varphi_0(x),$$

$$u_t(x,0) = \sum_{i=1}^{\infty} F_i(x) G_i'(0) = \sum_{i=1}^{\infty} F_i(x) H_{i0} = \sum_{i=1}^{\infty} F_i(x) \varphi_{1i} = \varphi_1(x).$$

Пусть $\varphi_i = (\varphi_{0i}, \varphi_{1i})^T$, $e_1 = (1,0)^T$ – единичный вектор. Тогда:

$$G_i(t) = e_1^T K_i(t) \varphi_i.$$

Таким образом, получили общий вид решения краевой задачи

$$u(x,t) = \sum_{i=1}^{\infty} e_1^T K_i(t)\varphi_i \sin(\pi i x), \tag{5}$$

что позволит далее построить соответствующий функционал.

4. Построение функционала Ляпунова – Красовского. Теперь перейдём к построению функционала Ляпунова – Красовского v с заданной производной.

Интегрируя равенство (2), получим

$$v(\widehat{u}_T(\varphi)) - v(\widehat{u}_0(\varphi)) = -\int_0^T \|\widehat{u}_t(\varphi)\|_{L_2}^2 dt.$$

Так как $\widehat{u}_T(\varphi) \underset{T\to\infty}{\longrightarrow} 0$ в силу асимптотической устойчивости системы, то

$$v(\varphi) = \int_0^{\infty} \|\widehat{u}_t(\varphi)\|_{L_2}^2 dt.$$

Раскроем подынтегральное выражение:

$$v(\varphi) = \int_0^{\infty}\int_0^1 u^2(y,t,\varphi)dydt. \tag{6}$$

Подставив решение (5) в равенство (6), получим

$$v(\varphi) = \int_0^{\infty}\int_0^1 \left(\sum_{i=1}^{\infty} e_1^T K_i(t)\varphi_i \sin(\pi i y)\right) \times \\ \times \left(\sum_{j=1}^{\infty} e_1^T K_j(t)\varphi_j \sin(\pi j y)\right) dydt.$$

В связи с ортонормированностью системы синусов переходим к следующей форме:

$$v(\varphi) = \sum_{i=1}^{\infty} \int_0^1 \sin^2(\pi i y)dy \int_0^{\infty} \varphi_i^T K_i^T(t) e_1 e_1^T K_i(t)\varphi_i dt. \tag{7}$$

Вычислим

$$\int_0^1 \sin^2(\pi i y)dy = \frac{1}{2} - \frac{\sin(2\pi i)}{4\pi i} = \frac{1}{2}.$$

Введём обозначение

$$U_i = \int_0^\infty 2K_i^T(t)e_1e_1^TK_i(t)dt.$$

Решив матричное уравнение Ляпунова $A_i^TU_i + U_iA_i = -2e_1e_1^T$, найдём матрицу

$$U_i = \begin{pmatrix} \frac{b}{s_i} - \frac{1}{b} & -\frac{1}{s_i} \\ -\frac{1}{s_i} & \frac{1}{bs_i} \end{pmatrix} = \begin{pmatrix} \frac{b}{s_i} & -\frac{1}{s_i} \\ -\frac{1}{s_i} & \frac{1}{bs_i} \end{pmatrix} + \begin{pmatrix} -\frac{1}{b} & 0 \\ 0 & 0 \end{pmatrix} = V_i - \frac{e_1e_1^T}{b}, \quad (8)$$

где $s_i = -a^2\pi^2i^2 + c$.

Подставив представление (8) в функционал (7), получим

$$v(\varphi) = \sum_{i=1}^{\infty} \frac{1}{4}\varphi_i^T \left(V_i - \frac{e_1e_1^T}{b} \right) \varphi_i = \sum_{i=1}^{\infty} \left(\frac{1}{4}\varphi_i^T V_i \varphi_i - \frac{\varphi_{0i}^2}{4b} \right). \quad (9)$$

Используя интегральное представление φ_i, перепишем равенство (9) в виде

$$v(\varphi) = \sum_{i=1}^{\infty} \left(\int_0^1 \int_0^1 \varphi^T(x_1)\sin(\pi i x_1)dx_1 V_i \varphi(x_2)\sin(\pi i x_2)dx_2 - \frac{\varphi_{0i}^2}{4b} \right).$$

Применив теорему Парсеваля, представим функционал в виде

$$v(\varphi) = \sum_{i=1}^{\infty} \int_0^1 \int_0^1 \varphi^T(x_1)\sin(\pi i x_1)dx_1 V_i \varphi(x_2)\sin(\pi i x_2)dx_2 - \\ - \int_0^1 \frac{\varphi_0^2(x)}{2b}dx.$$

Введём матрицу, которую будем называть матрицей Ляпунова для нашей краевой задачи:

$$U(x_1, x_2) = \sum_{i=1}^{\infty} \sin(\pi i x_1)\sin(\pi i x_2)V_i. \quad (10)$$

Получим окончательный вид функционала Ляпунова – Красовского:

$$v(\varphi) = \int_0^1 \int_0^1 \varphi^T(x_1)U(x_1, x_2)\varphi(x_2)dx_2dx_1 - \int_0^1 \frac{\varphi_0^2(x)}{2b}dx.$$

5. Проверка абсолютной сходимости ряда (10)**.** Для проверки абсолютной сходимости ряда для матрицы Ляпунова был проведён численный эксперимент в прикладном пакете MATLAB.

Рассмотрим сумму

$$\sum_{i=1}^{\infty} \|\sin(\pi i x_1)\sin(\pi i x_2) V_i\| .$$

Так как $|\sin(\pi i x_1)\sin(\pi i x_2)| \leqslant 1$, абсолютная сходимость исходного ряда следует из абсолютной сходимости ряда $\sum\limits_{i=1}^{\infty} \|V_i\|$.

В рамках исследования использовались параметры $a = 1$, $b = -2$, $c = 5$, при которых система является асимптотически устойчивой. На рис. 1 приведено поведение частичных сумм $\sum\limits_{i=1}^{N} \|V_i\|$ при $N = 500$.

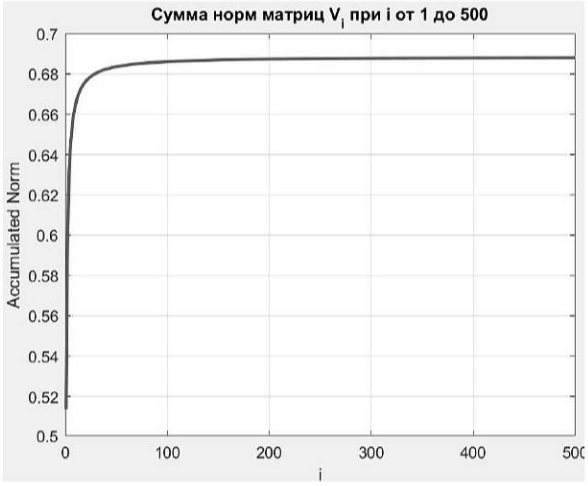


Рис. 1. Частичные суммы ряда (10)

Из рис. 1 следует, что по мере увеличения i сумма норм быстро растёт в начальный момент, но затем выходит на плато, что свидетельствует об абсолютной сходимости ряда V_i, а следовательно, и указывает на абсолютную сходимость соответствующего ряда для матрицы Ляпунова.

6. Заключение. В данной работе рассмотрено волновое уравнение, для решения которого применён метод Фурье. На основе полученного ряда построен функционал Ляпунова – Красовского с заданной производной вдоль решений системы, что в дальнейшем позволит исследовать устойчивость и робастную устойчивость системы.

Проведённый численный эксперимент подтвердил абсолютную сходимость ряда для матрицы Ляпунова при выбранных параметрах. В дальнейшем планируется рассмотреть волновое уравнение с запаздыванием по времени.

Литература

1. Kharitonov V. L. Time-Delay Systems: Lyapunov Functionals and Matrices. Boston, MA: Birkhauser, 2013. 322 p.

2. Харитонов В. Л. Функционалы Ляпунова с заданной производной. I. Функционалы полного типа // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2005. Т. 1. № 1. С. 110–117.

3. Rodriguez F., Roales M., Martin J. A. Exact solutions and numerical approximations of mixed problems for the wave equation with delay // Applied Mathematics and Computation. 2012. Vol. 219. No 6. P. 3178–3186.

4. Jornet M. Exact solution to a multidimensional wave equation with delay // Applied Mathematics and Computation. 2021. Vol. 409. P. 1–8.

УДК 519.712.1

Эстрада Хименес О.

Алгоритм определения последовательности движений, необходимых для изменения положения штабелируемых контейнеров

Рекомендовано к публикации доцентом Гришкиным В. М.

1. Введение. Контейнерные перевозки – один из наиболее распространенных видов грузоперевозок. Доставка грузов судами иногда является практически единственным способом перемещения различных грузов. Для этого используются транспортные контейнеры, представляющие собой транспортные единицы многоразового использования, предназначенные для временной перевозки и хранения грузов. Эти контейнеры оснащены механизмами, облегчающими их автоматизированную установку и снятие с транспортных средств.

Процесс перемещения транспортного контейнера включает в себя несколько этапов: от обработки груза до приема и отгрузки товаров в местах, известных как контейнерные пункты или терминалы. Основными функциями каждого морского контейнерного терминала являются погрузка, постановка, хранение и прием.

Процесс хранения контейнера включает его размещение в признанном и задокументированном месте, чтобы его можно было извлечь при необходимости. Контейнеры распознаются и организованы для максимальной эффективности загрузки. Краны, погрузчики, грузоподъемные устройства и т. п. являются основными технологическими средствами, применяемыми при обработке, погрузке и хранении контейнеров [1].

2. Постановка задачи. Обычно, скорость контейнерных перевозок относительно медленная по сравнению с другими видами грузоперевозок, на это могут влиять такие факторы, как скорость судна и время, затраченное на погрузочно-разгрузочные работы. Важно отметить, что модернизированные контейнерные терминалы, оснащенные современной техникой, позволяют существенно ускорить все

Эстрада Хименес Октавио – магистрант, Санкт-Петербургский государственный университет; e-mail: octavioestradajimenez@yandex.ru, тел.: +7(911)726-66-71

вышеперечисленные процессы. Данные терминалы являются базовыми в процессе перевозки грузов, и факт изучения их модернизации представляет собой важную цель исследования.

Очевидно, что для этой цели требуется специальное оборудование, которое значительно упростило бы обработку грузовых контейнеров и не отнимало бы большого количества рабочего времени. Другим, более простым подходом было бы рациональное планирование перевозок и организация грузовых операций. При планировании погрузки и разгрузки контейнеров в портах такое решение возможно только при предварительном компьютерном моделировании [2].

Основным предметом исследования является разработка алгоритма обработки морских контейнеров, способного сортировать продукцию в контролируемой среде, подобно условиям, в которых происходит операция хранения контейнеров в морском порту.

3. Основы алгоритма. Прежде всего, необходимо определить размеры рабочего пространства. При хранении морских контейнеров наиболее часто используемым оборудованием являются козловые краны на резиновых шинах, которые размещают контейнеры рядами и столбцами, расположенными поверх других. Эту форму расположения можно выразить как вектор (в информатике) матриц, а матрица аналогично является вектором векторов. Например, для размера $3 \times 3 \times 3$, представлен первый элемент вектора на рис. 1.

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

Рис. 1. Пример вектора векторов 3×3

Первый итератор вектора векторов представляет его положение относительно столбцов контейнеров, а второй итератор – его положение относительно рядов контейнеров, аналогично декартовой плоскости, где столбцы представляют координаты X, а ряды – координаты Y.

Третий итератор матричного вектора представляет его положение относительно контейнеров, соответствующее в декартовой плоскости координатам Z. Самый простой способ представить себе рабочее пространство – это вид сверху.

На рис. 2 единицы и нули обозначают занято место или нет. Это расположение назовем тензором памяти. Если посчитать доступные

$$\begin{bmatrix}1 & 0 & 0\\ 1 & 1 & 0\\ 0 & 1 & 0\end{bmatrix}\begin{bmatrix}0 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}0 & 0 & 0\\ 0 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}0 & 0 & 0\\ 0 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}$$

Рис. 2. Пример вектора матриц $4 \times 3 \times 3$

пустые места по координате Z, то получим матрицу, которую назовем матрицей доступного пространства (см. рис. 3).

$$\begin{bmatrix}3 & 4 & 4\\ 3 & 2 & 4\\ 4 & 3 & 4\end{bmatrix}$$

Рис. 3. Пример матрицы доступного пространства 3×3

Для обеспечения маневренности при перемещении контейнеров всегда будет учитываться свободное пространство или свободный уровень наверху.

4. Предпочтение движения. Теперь, можно разработать матрицу, которая будет определять предпочтительность движения контейнеров из-за маневренности кранов с резиновыми шинами и штабелирования контейнеров. Единственные основные аспекты, на которых следует сосредоточиться, это то, сколько места доступно в координате Z, и определение матрицы предпочтения движения, аналогичной понятию взвешенного графа в теории графов [3].

Например, для циклов итераторов $i = 2$, $j = 2$ и положения $x = 1$, $y = 1$ получим матрицу, представленную на рис. 4.

$$\begin{bmatrix}4 & 3 & 4\\ 1 & 0 & 1\\ 4 & 3 & 4\end{bmatrix}$$

Рис. 4. Пример матрицы предпочтения движения

Эта матрица предпочтений может быть построена с помощью двух циклов итераторов, которые зависят от размера рабочего пространства $(r \times c)$ и положения контейнера (x, y) (см. рис. 5).

Для значения матрицы на рис. 4, есть предпочтение переместить контейнер в любую сторону от его исходного положения, а не в другой ряд, поскольку так движется козловый кран на резиновых шинах. Считается, что проще сделать троллейное движение, верхнюю

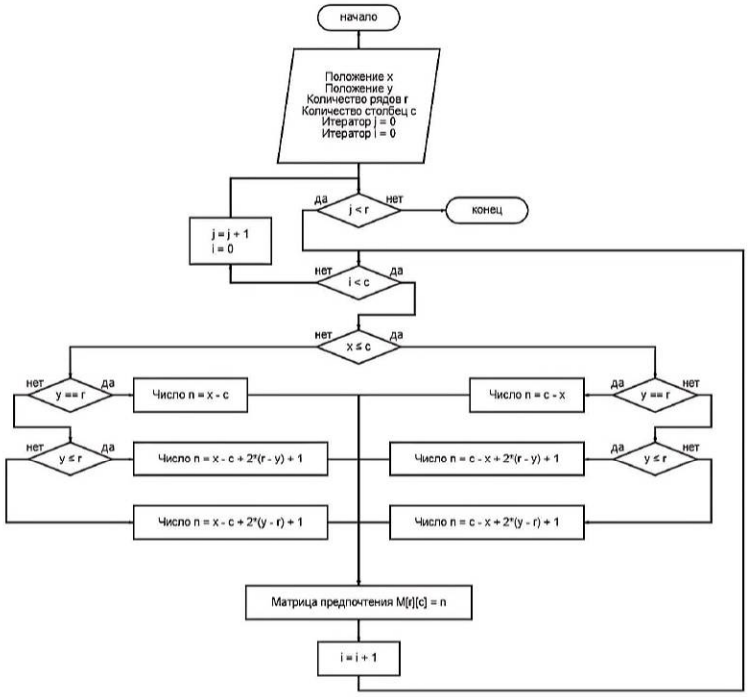


Рис. 5. Блок-схема алгоритма матрицы предпочтения

часть крана, передвигающуюся вокруг моста крана, чем козловый кран на резиновых шинах, предполагающее перемещение всего крана.

Имея предпочтения в отношении матрицы движения и матрицы доступного пространства, можно определить последовательность движения, которая временно очистит путь желаемого контейнера. Кроме того, можно определить количество шагов, необходимых для очистки пути, сравнивая точку отправления и точку назначения с доступной матрицей пространства. Допустим, если нужно переместить контейнер из положения $(1, 1, 0)$ и затем в пункт назначения $(0, 2, 0)$ из примера на рис. 2, то потребуется сделать два шага от положения и один шаг из пункта назначения, чтобы переместить контейнер в пункт назначения.

5. Последовательность движений. Теперь, можно разработать различные условия в зависимости от количества шагов, чтобы в целом определить последовательность. Каждое перемещение контейнера состоит из следующих шагов: очистка, перемещение, обмен и реорганизация. Очистка заключается в перемещении контейнера, который представляет собой препятствие для перемещения желаемого контейнера к месту назначения. Эта очистка может происходить как в месте положения контейнера, так и в пункте его назначения.

Перемещение – это самый простой из шагов. Он заключается лишь в доставке желаемого контейнера в пункт назначения. Этот шаг также можно использовать для введения или вывода контейнеров рабочей области или зоны хранения. Шаг обмена тесно связан с этапами перемещения, поскольку он происходит только в том случае, если точка назначения занята и контейнер, находившийся в точке назначения, займет место контейнера из точки положения. Это было сделано для поддержания порядка при перемещении контейнеров.

Наконец, будет проведена реорганизация, заключающаяся в возвращении временно перемещенного по очистке контейнера в исходное положение. Если воспользоваться примером перемещения контейнера из положения $(1, 1, 0)$ в пункт назначения $(0, 2, 0)$ на рис. 2, то получим следующие последовательности от источника к месту назначения соответственно:

- Очистка пункта положения $(1, 1, 1) > (0, 1, 1)$.
- Очистка пункта назначения $(0, 2, 0) > (1, 2, 0)$.
- Перемещение $(1, 1, 0) > (0, 2, 0)$.
- Обмен $(1, 2, 0) > (1, 1, 0)$.
- Реорганизация $(0, 1, 1) > (1, 1, 1)$.

Важно отметить, что есть особые случаи, требующие дополнительных действий. Один из таких случаев – когда положение и пункт назначения отличаются только координатой Z. В этих случаях последовательность делится на две последовательности. В первом случае самый высокий контейнер, какое бы ни было положение или пункт назначения, выделяется на свободное пространство, а во второй последовательности этот контейнер заменяется на самый нижний контейнер и реорганизуется с дополнительным шагом с запоминанием положения «самый высокий контейнер».

6. Упорядочить по приоритету. Некоторая концепция, которая будет полезной, заключается в том, чтобы упорядочить контейнер по приоритету, причем номер один будет наивысшим приоритетом, а полная вместимость места хранения – наименьшим приоритетом. Каждый раз при добавлении контейнера можно присвоить ему номер приоритета. Очевидно, что номера не могут повторяться, например, из рис. 2 получаем результат, представленный на рис. 6.

$$\begin{bmatrix} 1 & 0 & 0 \\ 6 & 27 & 0 \\ 0 & 3 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 15 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Рис. 6. Пример вектора матриц с номерами приоритета

Принимая эти номера за основу для положений можно создать функцию, которая сравнивает каждое из чисел со статическим вектором матриц с номерами приоритета по порядку (см. рис. 7).

$$\begin{bmatrix} 25 & 26 & 27 \\ 22 & 23 & 24 \\ 19 & 20 & 21 \end{bmatrix} \begin{bmatrix} 16 & 17 & 18 \\ 13 & 14 & 15 \\ 20 & 11 & 12 \end{bmatrix} \begin{bmatrix} 7 & 8 & 9 \\ 4 & 5 & 6 \\ 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} 36 & 36 & 36 \\ 36 & 36 & 36 \\ 36 & 36 & 36 \end{bmatrix}$$

Рис. 7. Пример вектора матриц с номерами приоритета

Последний уровень заполнен одинаковыми номерами, это просто означает, что он недоступен. Чтобы провести сравнение, необходимо отредактировать исходный вектор матриц с номерами приоритета, назначив номера временного приоритета в порядке убывания от номера с наименьшим приоритетом (см. рис. 8).

$$\begin{bmatrix} 23 & 0 & 0 \\ 25 & 27 & 0 \\ 0 & 24 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 26 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Рис. 8. Пример вектора матриц с номерами приоритета

Теперь видно, куда должен идти каждый контейнер, но какой из них следует переместить первым? Тот, у которого самая короткая последовательность. Внутри функции можно перебирать все возможные движения и выбирать одно с самой короткой последовательностью. Также важно помнить, что выбранный контейнер должен находиться на самом нижнем неупорядоченном уровне, иначе может

быть создан бесконечный цикл. Это означает, что самый низкий уровень всегда будет упорядочен первым, а затем будет следовать за следующим по порядку уровнем. Имея эти критерии, на рис. 6 можно переместить либо контейнер номер 27, либо контейнер 3.

7. Заключение. Разработка подобных алгоритмов представляет собой простой начальный этап академического исследования возможной автоматизации задач в порту терминала. Для большинства этих задач уже существуют средства автоматизации, эта автоматизация могла бы существенно модернизировать и улучшить не только портовые терминалы, но и всю сферу логистики и транспорта. Обработка грузов могла бы не требовать особых усилий, или же это могла бы быть машина, работающая безостановочно.

В данной работе объяснены основные концепции, которые были приняты во внимание при построении алгоритма. Одним из недостатков этого алгоритма является то, что он работает в полную силу только тогда, когда количество столбцов и строк равно или больше двух, и когда свободное пространство сверху для маневренности более чем в два раза превышает согласованную по Z вместимость.

Алгоритм разработан на C++, в общей сложности потребовалось около 4 тысяч строк кода [4]. Этот алгоритм нельзя назвать существенным результатом в модернизации транспорта и логистики, и, конечно в нём многое можно улучшить. Однако, он может дать вдохновение для новых идей применения автоматизации в различных задачах, особенно с многочисленными повторениями действий.

Литература

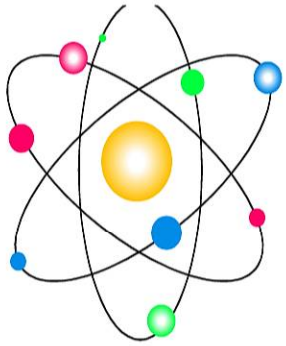
1. Братусь О. Ю., Иванцова Ю. А. Оптимизация процесса грузоперевозок: контейнерный терминал // E-Scio. 2020. Т. 12. № 51. С. 608–620.

2. Устинов А. В., Днепровский В. В. Анализ критериев составления грузового плана морского судна-контейнеровоза // Приазовский государственный технический университет. 2005. Т. 15. № 1. С. 191–194.

3. Кристофидес Н. Теория графов. Алгоритмический подход. М.: Мир, 1978. 432 с.

4. RTG-Algorithm [Электронный ресурс]: `URL:https://github.com/YOEstradaJimenez/RTG-Algorithm` (дата обращения: 12.03.2024).

2. Математические методы в механике и физике



УДК 519.71

Вавренюк В. В.

Нечеткий регулятор в задаче стабилизации солнечного паруса

Рекомендовано к публикации доцентом Шиманчуком Д. В.

1. Введение. Идея полетов в космосе с использованием солнечного паруса (СП) принадлежит советскому ученому Фридриху Цандеру, который сформулировал эту мысль в 20-е годы XX века. Фотоны (частицы солнечного света) создают давление, передавая имеющийся у них импульс поверхности, на которую попадает свет. Таким образом реализуется движение СП под действием давления частиц света (фотонов), и, соответственно, ракетное топливо тратиться не будет, а при изменении угла расположения конструкции СП можно менять направление вектора тяги. Идея использования солнечного паруса для удержания космического аппарата (КА) в окрестности коллинеарной точки либрации была сформулирована в работе Буклесса, Макиннеса [2]. Под солнечным парусом в данной статье будем подразумевать КА с установленной и развернутой на нем зеркальной поверхностью, отражающая способность которой позволяет сообщать КА управляющее ускорение вследствие давления света. Управление СП в данной работе будет производиться с помощью регулятора на основе нечеткой логики, в области которой в настоящее время активировались исследования применительно к управлению техническими объектами. Данный тип регуляторов хорошо зарекомендовал себя в управлении сложными нелинейными системами, а также системами с нелинейными внешними возмущениями. Движение КА будем рассматривать в рамках круговой ограниченной задачи трех тел [1] – модели небесной механики, которая описывает движения тела бесконечно малой массы относительно двух притягивающих центров, взаимодействующих друг с другом по закону всемирного тяготения. Малое тело не влияет на движение притягивающихся центров, которые имеют соизмеримые массы и движутся относительно своего центра масс по круговым орбитам. Частным

Вавренюк Владислав Владимирович – магистрант, Санкт-Петербургский государственный университет; e-mail: st076914@spbu.student.ru, тел.: +7(914)224-82-26

случаем решения круговой ограниченной задачи трех тел являются точки либрации (точки Лагранжа, L-точки) [1], их расположение показано на рис. 1.

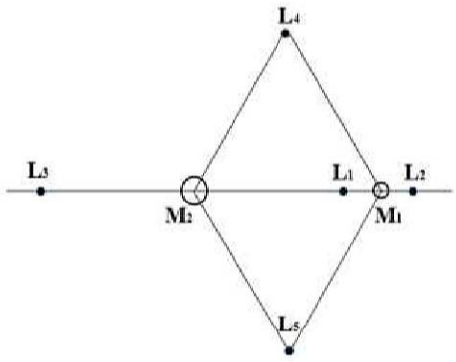


Рис. 1. Точки либрации

2. Постановка задачи. Пусть космический аппарат, оснащенный солнечным парусом с заданными характеристиками (m – масса, S – площадь паруса), движется в окрестности точки либрации L_1 системы Солнце – Земля в соответствии с математической моделью круговой ограниченной задачи трех тел при использовании хилловского приближения [3], [4]:

$$\begin{cases} \dot{x}_1 = x_2 + y_1, \\ \dot{x}_2 = -x_1 + y_2, \\ \dot{x}_3 = y_3, \\ \dot{y}_1 = -\frac{3x_1}{||x||^3} + 2x_1 + y_2 + u_1, \\ \dot{y}_2 = -\frac{3x_2}{||x||^3} - x_2 - y_1 + u_2, \\ \dot{y}_2 = -\frac{3x_3}{||x||^3} - x_3 + u_3, \end{cases} \tag{1}$$

где $\overline{x} = (x_1, x_2, x_3)$ – вектор координат центра масс КА, $\overline{y} = (y_1, y_2, y_3)$ – вектор скоростей КА, $\overline{u} = (u_1, u_2, u_3)$ – вектор управляющего воздействия со стороны СП, $\|\cdot\|$ – евклидова норма. Уравнения представлены во вращающейся системе координат $Ox_1x_2x_3$,

где O – центр инерции Земли, ось Ox_1 направлена вдоль оси, соединяющей O и центр масс Солнца. В принятой модели за единицу расстояния принимается величина, равная 10^{-2} астрономические единицы ?1,5 $\times$ 10^6 км, за единицу времени – год, деленный на 2π, что примерно равно 58 суткам, за единицу скорости – 298,659 м/с, и за единицу ускорения – 5,94649 $\times$ $10^{-5}/^2$. Точка либрации L_1 во вращающейся системе неподвижна и имеет координаты $x^* = (1, 0, 0)$ и $y^* = (0, 1, 0)$. Неуправляемая система (1) является гамильтоновой, где функция Гамильтона [3]

$$H = \frac{||\overline{y}||^2}{2} - \frac{3}{||\overline{x}||} - \frac{3}{2}x_1^2 + \frac{||\overline{x}||^2}{2} + x_2y_1 - x_1y_2$$

сохраняет свое значение на траекториях движения, из-за чего является интегралом системы (1). Рассмотрим линеаризованный вид неуправляемой системы (1) в окрестности точки L_1 [3]

$$\begin{cases} \dot{x}_1 = x_2 + y_1, \\ \dot{x}_2 = -x_1 + y_2, \\ \dot{x}_3 = y_3, \\ \dot{y}_1 = 8(x_1 - 1) + (y_2 - 1), \\ \dot{y}_2 = -4x_2 - y_1, \\ \dot{y}_2 = -4x_3, \end{cases} \tag{2}$$

для которой известны собственные значения [3]

$$\lambda_{1,2} = \pm\sqrt{1 + 2\sqrt{7}}, \quad \lambda_{3,4} = \pm i\sqrt{2\sqrt{7} - 1}, \quad \lambda_{5,6} = \pm 2i.$$

Из положительности собственного значения $\lambda_1 = \sqrt{1 + 2\sqrt{7}}$ следует неустойчивость точки либрации L_1 и необходимость управляющего воздействия для стабилизации КА в окрестности данной точки.

3. Функция опасности. Функция опасности [5–7] – функция фазовых переменных $\overline{x}$ и $\overline{y}$, служащая для определения удаленности КА от точки либрации L_1. Данная функция является важной

характеристикой поведения решения как линеаризованной системы (2), так и системы (1) в окрестности коллинеарной точки либрации L_1 и имеет вид

$$d_1 \triangleq \overline{a}_1 \overline{l},$$

где $\overline{a}_1 = (a_{11}, a_{12}, 0, a_{14}, a_{15}, 0,)$ – собственный вектор-строка, который соответствует собственному значению $\lambda_1 = \sqrt{1 + 2\sqrt{7}}$ линеаризованной системы (2), а $\overline{l} = (x_1 - 1, x_2, x_3, y_1, y_2 - 1, y_3)^{\intercal}$ – вектор-столбец. Вектор $\overline{a}_1$ определяется с точностью до множителя, для определенности можно положить [6]

$$a_{14}^2 + a_{15}^2 = 1, \tag{3}$$

$$a_{11} > 0.$$

В силу неуправляемой линеаризованной системы (2) получим

$$\dot{d_1} = \lambda_1 d_1. \tag{4}$$

Решение уравнения (4) имеет вид

$$d_1 = c_1 e^{\lambda_1 t},$$

где c_1 – некоторая константа, определяемая начальными условиями. Известно [6], что в общем случае для системы (2) знак функции опасности определяет направление движения КА из окрестности L_1. Так, с учетом условия (3), значение $d_1 > 0$ соответствует направлению на Солнце, $d_1 < 0$ – на Землю. Таким образом, чтобы КА находился в некоторой окрестности L_1 длительное время, необходимо обеспечить выполнение условия ограниченности d_1 в окрестности этой точки:

$$d_1^{\min} \leqslant d_1 \leqslant d_1^{\max},$$

где $d_1^{\min}$ и $d_1^{\max}$ – достаточно малые величины.

Для управляемой линеаризованной системы

$$\begin{cases} \dot{x}_1 = x_2 + y_1, \\ \dot{x}_2 = -x_1 + y_2, \\ \dot{x}_3 = y_3, \\ \dot{y}_1 = 8(x_1 - 1) + (y_2 - 1) + u_1, \\ \dot{y}_2 = -4x_2 - y_1 + u_2, \\ \dot{y}_2 = -4x_3 + u_3, \end{cases}$$

поведение функции опасности d_1 описывается уравнением

$$\dot{d_1} = \lambda_1 d_1 + a_{14} u_1 + a_{15} u_2. \tag{5}$$

4. Модель КА с солнечным парусом. Солнечный парус представляет собой зеркальную поверхность определенной площади S массой m, обладающей отражательными свойствами. На солнечный парус действует давление солнечного света, равное $P = 4{,}56 \times 10^{-6}$ Па. Фотоны солнечного света дважды обмениваются импульсом с поверхностью паруса: при сталкивании с ним и при отражении; а после парус передает воздействие всему КА.

4.1. Модель солнечного паруса. Пусть КА с СП движется таким образом, чтобы орт вектора нормали паруса был всегда параллелен плоскости эклиптики, из-за чего и управляющее ускорение будет лежать в плоскости, параллельной плоскости эклиптики. Угол φ – угол между вектором нормали солнечного паруса $\overline{n}$ и ортом направления от Земли к Солнцу $\overline{i}$. Данный угол $\varphi \in [-\pi/2, \pi/2]$ и в граничных значениях $\varphi = -\pi/2$ и $\varphi = \pi/2$ парус будет параллелен потоку солнечных лучей, вследствие чего импульсов от фотонов и, соответственно, управляющего воздействия не будет. Меняя значение угла φ, можно менять направление вектора силы тяги, из-за чего можно менять и положение СП в пространстве. На рис. 2 обозначены силы, действующие на солнечный парус, и векторы, заданные ранее. Тут точки M_1 и M_2 – точки нахождения Земли и Солнца, а векторы $\overline{F}_1$ и $\overline{F}_2$ – поглощающая и отражающая силы, соответственно, при этом $\overline{F}_1$ сонаправлена с потоком солнечных лучей $\overline{F}_1 \uparrow\uparrow -\overline{i}$. Данные силы можно найти по формулам

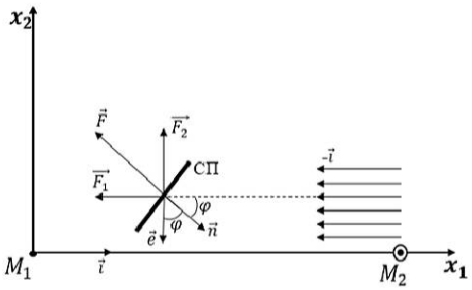


Рис. 2. Механика СП

$$\overline{F}_1 = -PS\left(-\bar{i}, \overline{n}\right)\bar{i},$$

$$\overline{F}_2 = -PS\left(-\bar{i}, \overline{n}\right)\bar{e},$$

где $\left(-\bar{i}, \overline{n}\right)$ – скалярное произведение. Тогда итоговая сила тяги $\overline{F}$, действующая на СП, находится как

$$\overline{F} = -2PS\left(-\bar{i}, \overline{n}\right)^2 \overline{n}.$$

Спроецировав данный вектор на оси M_1x_1, M_1x_2, M_1x_3, получим управляющее воздействие [6]:

$$\begin{cases} u_1 = -2\frac{PS}{am}\cos^3\varphi, \\ u_2 = -2\frac{PS}{am}\cos^2\varphi\sin\varphi, \\ u_3 = 0, \end{cases} \tag{6}$$

где a – единица ускорения в принятой системе единиц.

4.2. Ориентация солнечного паруса. Изменяя угол наклона солнечного паруса, можно изменить и направление вектора силы тяги, из-за чего будет меняться положение КА в пространстве. Из формул (5), (6) и (3) получим [6]

$$\dot{d}_1 = \lambda_1 d_1 - 2a_{14}\frac{PS}{am}\cos^3\varphi - 2a_{15}\frac{PS}{am}\cos^2\varphi\sin\varphi,$$

$$\dot{d_1} = \lambda_1 d_1 - 2\frac{PS}{am} f(\varphi),$$

где $f(\varphi) = \cos^2\varphi \sin(\varphi + \widehat{\varphi})$, $\widehat{\varphi} = \arctan\frac{\lambda_1^2+3}{2\lambda_1}$. Для удержания СП в области $d_1 > 0$ (движение на Солнце) необходимо, чтобы выполнялось неравенство

$$\lambda_1 d_1 - 2\frac{PS}{am} f(\varphi^*) < 0, \tag{7}$$

а для удержания в области $d_1 < 0$ (движение на Землю):

$$\lambda_1 d_1 - 2\frac{PS}{am} f(\varphi_*) > 0, \tag{8}$$

где φ^* и φ_* – максимальный и минимальный допустимые углы (границы) ориентации солнечного паруса, которые можно найти из наибольшего и наименьшего значения (экстремумов) функции $f(\varphi)$. Для этого найдем

$$\frac{df(\varphi)}{d\varphi} = \cos\varphi\,(\cos\varphi\cos(\varphi + \widehat{\varphi}) - 2\sin\varphi\sin(\varphi + \widehat{\varphi})) \tag{9}$$

и построим график функции $f(\varphi)$ (рис. 3). Приравнивая уравнение (9) к нулю, можно найти значения углов из $\varphi \in [-\pi/2; \pi/2]$, и рассмотрев график функции $f(\varphi) = \cos^2\varphi \sin(\varphi + \widehat{\varphi})$, получим

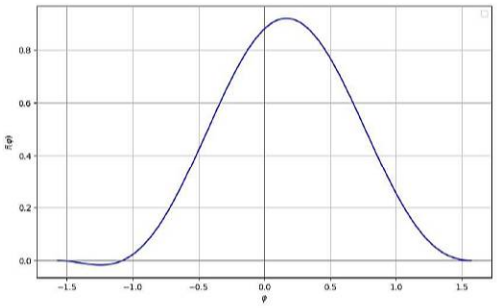


Рис. 3. График $f(\varphi) = \cos^2\varphi \sin(\varphi + \widehat{\varphi})$

$$\varphi^* = 0{,}168016, \tag{10}$$

$$\varphi_* = -1{,}24375. \tag{11}$$

Тогда из неравенств (7), (8), учитывая значения (10), (11), можно получить оценку снизу и сверху для значения функции опасности, а, следовательно, и оценку области управляемости:

$$d_1^{\min} = \frac{2PS\cos^2\varphi_* \sin(\varphi_* + \widehat{\varphi})}{\lambda_1 am},$$

$$d_1^{\max} = \frac{2PS\cos^2\varphi^* \sin(\varphi^* + \widehat{\varphi})}{\lambda_1 am}.$$

5. Построение закона управления на базе нечеткой логики. Контролировать положение КА в пространстве предлагается с помощью угла ориентации солнечного паруса, а оценивать это положение можно по функции опасности. Таким образом, необходимо построить зависимость изменения угла наклона паруса φ от функции опасности d_1. Для этого зададим значение функции опасности, относительно которого предполагается удержание КА:

$$\overline{d_1} = \frac{d_1^{\min} + d_1^{\max}}{2},$$

и введем линейные функции принадлежности [8], показывающие степень приближения значения d_1 к $d_1^{\min}$ и $d_1^{\max}$ соответственно (рис. 4):

$$f_l(d_1) = \begin{cases} 1, & \text{если } d_1 < d_1^{\min}, \\ \frac{\overline{d_1} - d_1}{\overline{d_1} - d_1^{\min}}, & \text{если } d_1^{\min} \leqslant d_1 \leqslant \overline{d_1}, \\ 0, & \text{если } d_1 > \overline{d_1}, \end{cases}$$

$$f_h(d_1) = \begin{cases} 1, & \text{если } d_1 > d_1^{\max}, \\ \frac{\overline{d_1} - d_1}{\overline{d_1} - d_1^{\max}}, & \text{если } \overline{d_1} \leqslant d_1 \leqslant d_1^{\min}, \\ 0, & \text{если } d_1 < \overline{d_1}. \end{cases}$$

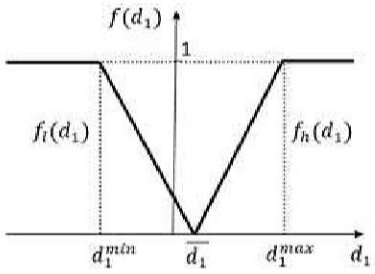


Рис. 4. Общая функция принадлежности

Чтобы движение СП в окрестности точки либрации L_1 стабилизировалось, угол наклона паруса должен находиться в пределах либо $\varphi \in [\varphi_*, \varphi^*]$, либо $\varphi \in [\varphi^*, \frac{\pi}{2}]$ (см. рис. 3 и значения (10), (11)). Тогда дискретный режим изменения управляющего параметра в зависимости от функции опасности для нелинейной модели (1) при условии $\varphi_* \leqslant \varphi \leqslant \varphi^*$ запишется в виде

$$\varphi(i+1) = \varphi(i) - \Delta\varphi f_l(d_1) + \Delta\varphi f_h(d_1), \tag{12}$$

а для промежутка $[\varphi^*, \pi/2]$ получим

$$\varphi(i+1) = \varphi(i) + \Delta\varphi f_l(d_1) - \Delta\varphi f_h(d_1), \tag{13}$$

где $\Delta\varphi$ – заданное малое значение изменения угла ориентации солнечного паруса за один шаг.

6. Выводы. Для описания модели управляемого движения СП проанализированы математическая модель солнечного паруса и управляющего воздействия (6), сообщающие некоторое ускорение СП, зависящее от угла наклона солнечного паруса. Также на базе нечеткой логики построены два закона изменения угла наклона солнечного паруса для двух промежутков допустимых значений, т. е. построен нечеткий регулятор (12), (13) угла ориентации паруса в задаче стабилизации СП в окрестности точки либрации L_1.

Литература

1. Дубошин Г. Н. Небесная механика. Основные задачи и методы. М.: Наука, 1968. 798 с.

2. Bookless J., McInnes C. Control of Lagrange point orbits using solar sail propulsion // Acta Astronautica. 2008. Vol. 62. P. 159–176.

3. Шмыров В. А. Стабилизация управляемого орбитального движения космического аппарата в окрестности коллинеарной точки либрации L_1 // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2005. № 2. С. 193–199.

4. Шмыров А. С., Шмыров В. А., Шиманчук Д. В. Управление орбитальным движением космического аппарата вблизи точки либрации с помощью изменения отражательной способности солнечного паруса // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2024. Т. 20. № 2. С. 298–306.

5. Шиманчук Д. В., Шмыров В. А. Проблемы стабилизации орбитального движения космического аппарата в окрестности коллинеарной точки либрации // Молодой ученый. 2014. № 3(62). С. 20–24.

6. Шиманчук Д. В., Шмыров А. С., Шмыров В. А. Управляемое движение солнечного паруса в окрестности коллинеарной точки либрации // Письма в астрономический журнал. 2020. Т. 46. № 3. С. 193–200.

7. Шиманчук Д. В., Шмыров А. С. Построение траектории возвращения в окрестность коллинеарной точки либрации системы Солнце – Земля // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2013. Т. 9. № 2. С. 75–84.

8. Шиманчук Д. В. Построение нечеткого регулятора для удержания солнечного паруса вблизи коллинеарной точки либрации // XIV Всероссийское совещание по проблемам управления. М.: Институт проблем управления им. В. А. Трапезникова РАН. 2024. С. 1488–1492.

УДК 519.712.2

Викторов Б. В.

Алгоритм обхода группой роботов набора точек

Рекомендовано к публикации доцентом Шиманчуком Д. В.

1. Введение. В работе решена задача построения пути для группы роботов с учётом ограничения на расстояние между роботами. Это ограничение описывает дальность работы радиоканала, что имеет важное значение для задач децентрализованного и централизованного управления [1]. В реальном мире эта проблема возникает в задачах изучения и анализа местности.

2. Постановка задачи. Пусть есть двумерная евклидова плоскость, на которой определено множество точек A. Модели роботов описываются системой дифференциальных уравнений вида

$$\begin{cases} \dfrac{dx_i}{dt} = u_{i,1}(t), \quad i = \overline{1,2}, \\ \dfrac{dy_i}{dt} = u_{i,2}(t), \quad i = \overline{1,2}, \end{cases} \tag{1}$$

где $t > 0$ – время, (x_i, y_i) – координаты i-го робота, $u_{i,j}(t)$, $i = \overline{1,2}$, $j = \overline{1,2}$ – программное управление. Под посещением точки роботом понимается совпадение координат робота и точки. Цель – построить пути для группы роботов, двигаясь по которым они посетят все точки из множества A. При этом критерием качества может быть, например, время прохождения пути. Построим кусочно-линейные пути, которые задают перемещение робота от точки к точке, рассматривая соответствующие системы диффиренциальных уравнений с начальными условиями, задающими положение робота. Считаем, что взаимное расстояние между роботами не может быть больше заданного значения R: $(x_2 - x_1)^2 + (y_2 - y_1)^2 \leqslant R^2$. На каждом участке выбирается значение управления, которое для данного промежутка принимаем константой: $u_{i,1} = \text{const}, u_{i,2} = \text{const}$. На управление накладывается ограничение $\sqrt{u_{i,1}^2 + u_{i,2}^2} = u_0 = \text{const}$.

Викторов Борис Викторович – студент, Санкт-Петербургский государственный университет; e-mail: boris@victorov.com, тел.: +7(921)447-34-68

Учитывая вышесказанное, из системы (1) получим

$$\begin{cases} \Delta x_i = u_{i,1}\Delta t, & i = \overline{1,2}, \\ \Delta y_i = u_{i,2}\Delta t, & j = \overline{1,2}. \end{cases}$$

Тогда находим

$$\Delta r_i^2 = \Delta {x_i}^2 + \Delta {y_i}^2 = \Delta t^2 (u_{i,1}^2 + u_{i,2}^2).$$

Откуда следует, что перемещение

$$\Delta r_i = u_0 \Delta t.$$

Таким образом, можно сказать, что, минимизируя расстояние, мы также минимизируем время. Заметим, что представленная модель движения соответствует, например, модели голономного мобильного робота [2].

3. Наивный алгоритм. Блок-схема алгоритма представлена на рис. 1, где для упрощения записи введены следующие обозначения: T – ближайшая точка к текущей точке ведущего, T_l – текущая точка ведущего, T_f – текущая точка ведомого, T_p – одна из точек пересечения, X_l – текущее положение робота ведущего, X_f – текущее положение робота ведомого, C_1 – окружность с центром в T_l радиуса R, C_2 – окружность с центром в X_f и радиусом, равным расстоянию от ведущего до T_l. Данный алгоритм работает на простых геометрических принципах. Этот алгоритм, показывает существование решения.



Рис. 1. Блок схема алгоритма

4. Пример. Рассмотрим пример (см. рис. 2): тёмные точки (верхние две точки) – роботы, светлые – точки множества A. Слева изображены начальные данные, а справа – итог работы алгоритма. На рис. 3, 4 изображены итерации алгоритма, где жирная линия обозначает участок пути ведущего робота.

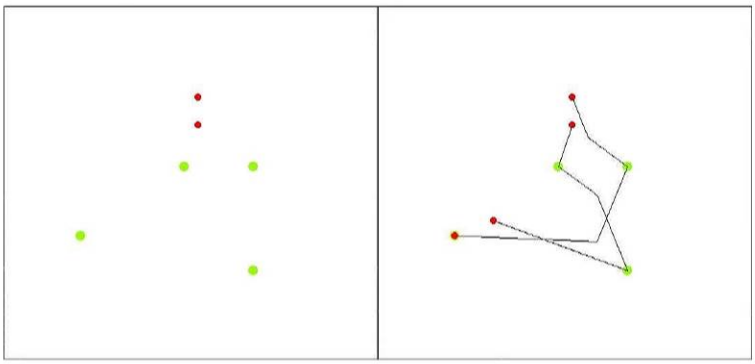


Рис. 2. Начальные данные и итоговый путь

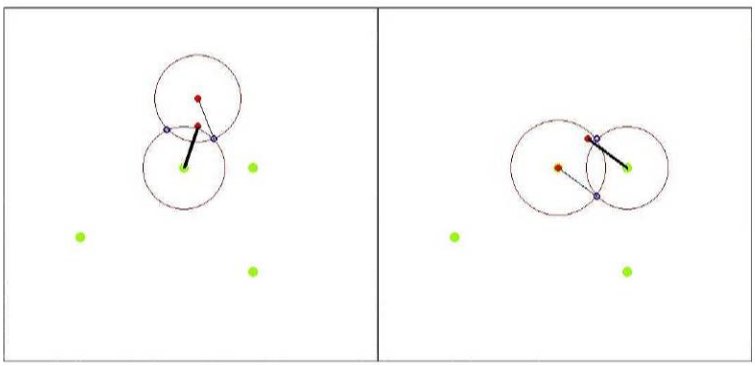


Рис. 3. Итерация 1 и 2

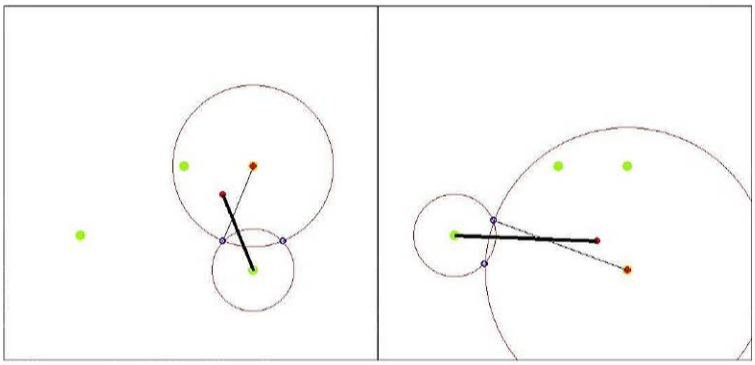


Рис. 4. Итерация 3 и 4

5. Заключение. Предложен наивный алгоритм для построения пути для группы роботов. Поскольку в алгоритме находятся пересечения двух окружностей и ведомый робот сначала едет в точку пересечения с меньшим расстоянием до T_f, то в дальнейшем предлагается вместо этого находить точку на дуге окружности C_2, ограниченной точками пересечения, с минимальным расстоянием до T_f. Эту точку можно находить, например, с помощью метода касательных [3].

Литература

1. Ярмолинский А. М., Викторов Б. В., Мещеряков А. Д., Глазов Ю. С. Мобильные роботы в международном стандарте RoboCup SSL: система управления // XIV Всероссийское совещание по проблемам управления (ВСПУ-2024). М.: ИПУ РАН, 2024. С. 1660–1664.

2. Ивкин К. А., Шиманчук Д. В. Автономное движение омниколесного робота по расчетной траектории // Процессы управления и устойчивость. 2020. Т. 7. № 1. С. 107–111.

3. Аббасов М. Э. Методы оптимизации. СПб.: Изд-во ВВМ, 2014. 64 с.

УДК 539.3

Вяльцева В. В.

Оценка наведенной анизотропии эффективных свойств материалов с трещинами

Рекомендовано к публикации профессором Прониной Ю. Г.

1. Введение. Микротрещины и другие дефекты изменяют эффективные свойства материалов [1, 2], например, электропроводность, упругость, теплопроводность или акустические свойства. Для характеризации случайным образом ориентированных круглых трещин радиусом r_i внутри объема V, Бристоу [3] ввел скалярный параметр плотности трещин

$$\rho_{3D} = \frac{1}{V}\sum_i r_i^3.$$

В случае наличия предпочтительной ориентации (ориентаций) в системе трещин, скалярный параметр необходимо заменить на тензорный [1]:

$$\boldsymbol{\rho} = \boldsymbol{\rho}_{3D} = \frac{1}{V}\sum_i (r^3 \boldsymbol{nn})_i, \tag{1}$$

где r и $\boldsymbol{n}$ – радиус и нормаль i-й трещины. Тогда, например, компоненты тензора упругой податливости определяются по формуле [1]:

$$\Delta S_{ijkl} = \frac{32(1-\nu_0^2)}{3(2-\nu_0)E_0}\left[\frac{1}{4}(\delta_{ik}\boldsymbol{\rho}_{jl} + \delta_{il}\boldsymbol{\rho}_{jk} + \delta_{jk}\boldsymbol{\rho}_{il} + \delta_{jl}\boldsymbol{\rho}_{ik}) - \frac{\nu_0}{2}\boldsymbol{\beta}_{ijkl}\right], \tag{2}$$

где $\boldsymbol{\beta} = \frac{1}{V}\sum_i (r^3 \boldsymbol{nnnn})_i$ – дополнительный тензор плотности трещин четвертого ранга. В этом случае наличие микротрещин делает изначально изотропный материал анизотропным.

Идентификация внутренних дефектов сопряжена с рядом технических сложностей и требует специального дорогостоящего оборудования.

Вяльцева Виктория Витальевна – магистрант, Санкт-Петербургский государственный университет; e-mail: valcevav@gmail.com, тел.: +7(952)892-77-38

Работа выполнена при финансовой поддержке РНФ, проект № 25-29-00565

Поэтому актуален вопрос оценки изменения эффективных свойств материалов по информации, которую можно получить с поверхности образцов. В работе [4] эта задача решена для изотропного случая (при случайном распределении ориентации трещин). В статье [5] были предложены формулы для оценки трехмерной плотности трещин по двумерной плотности следов этих трещин на поверхности в случае, когда трещины имеют определенную ориентацию. Целью данной работы является определение анизотропных эффективных упругих характеристик материала по следам трещин на гранях вырезанного из него образца.

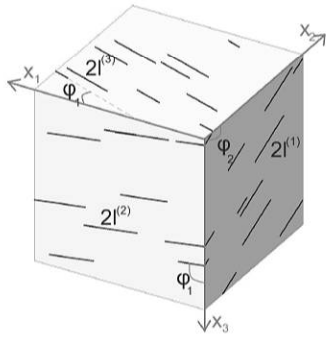


Рис. 1. Следы трещин на гранях образца

2. Постановка задачи. Рассматривается образец в форме параллелепипеда, ослабленный системой параллельных круглых плоских трещин радиусов r, центры которых случайным образом равномерно распределены по объему. Пусть система координат расположена так, что ее оси совпадают с тремя ребрами образца (рис. 1). Считаем известными длины следов трещин $2l_i$ на гранях образца с нормалями Ox_i и углы φ_i, образуемые следами трещин в плоскости Ox_ix_{i+1} с осью Ox_i.

Необходимо определить эффективные упругие свойства материала по данным о следах трещин на поверхности вырезанного из него образца.

3. Решение. Ориентация трещин определяется однозначно с помощью двух углов: α – угол между трещиной и осью x_3 и β – угол поворота трещины вокруг оси x_3, отсчитываемый от оси x_2. При этом

$$\alpha = \arcsin\left(\frac{\tan^2\varphi_2}{\cos^2\varphi_1} + 1\right)^{-\frac{1}{2}}, \quad \beta = \varphi_1 + \frac{\pi}{2}(1 + \operatorname{sign}\tan\varphi_1 + \operatorname{sign}\tan\varphi_2).$$

Плотности следов на гранях образца определяются по формуле

$$\boldsymbol{\rho}_{2D,i} = \frac{1}{S}\sum_k (l^2 \boldsymbol{nn})_k = \rho_{2D,i}(\boldsymbol{nn})_k, \tag{3}$$

где S – площадь грани образца, l и $\boldsymbol{n}$ – полудлина и нормаль k-го следа на грани с нормалью Ox_i.

Зная двумерную плотность следов, трехмерную плотность трещин определенной ориентации можно вычислить, например, по формуле [5]:

$$\rho_{3D} = \frac{3}{4\cos\alpha}\rho_{2D,3}. \tag{4}$$

Взятие среднего арифметического, полученного по каждой грани образца, существенно уточняет полученную оценку, которая используется для составления итоговых тензоров $\boldsymbol{\rho}$ и $\boldsymbol{\beta}$ и определения податливости по формуле (2).

4. Определение эффективных упругих свойств. В качестве примера рассмотрим изначально изотропный материал и покажем, как наличие системы параллельных трещин приводит к появлению анизотропии его свойств (когда модули Юнга в разных направлениях отличаются). Для анализа изменений эффективных свойств рассмотрим несколько систем трещин с разными ориентациями.

Площадь грани каждого модельного кубического образца составляет $S = 1$ см^2, количество трещин внутри образца $N = 200$, радиус трещин $r = 1$ мм, в качестве итогового значения плотности берется среднее арифметическое по $K = 100$ образцам (учитываются три грани образца). Изначальные модуль Юнга $E_0 = 160$ ГПа и коэффициент Пуассона $\nu_0 = 0{,}3$.

Случайным образом в каждом из K образцов задавали N трещин фиксированной ориентации. Точное значение плотности трещин $\boldsymbol{\rho}_{3D}$ для сопоставления с предсказанием определялось по формуле (1). Восстановленное значение плотности вычислялось с помощью соотношений (3) и (4) с последующим осреднением по всем образцам. После чего для точного и восстановленного значений плотности определялись эффективные упругие анизотропные свойства материала.

Полученные результаты представлены в таблице. В первом столбце записаны углы φ_1, φ_2, по которым определяется ориентация системы трещин. Следующие три столбца показывают значения эффективных модулей Юнга E_i, вычисленные по известным точным

значениям плотности трещин, а в последних трех столбцах – восстановленные модули Юнга E_i^*. Здесь E_i – модуль упругости в направлении оси Ox_i.

Таблица. Эффективные модули Юнга (ГПа)

φ_1, φ_2	E_1	E_2	E_3	E_1^*	E_2^*	E_3^*
$\frac{3\pi}{4}, \frac{3\pi}{4}$	117,50	117,50	117,50	117,29	117,29	117,29
$0, \frac{3\pi}{4}$	160,00	104,70	104,70	160,00	105,38	105,38
$\frac{\pi}{5}, \frac{\pi}{5}$	136,51	121,29	101,28	136,22	120,87	100,75
$\frac{\pi}{3}, \frac{\pi}{10}$	128,43	147,62	93,11	128,45	147,63	93,15
$\frac{\pi}{6}, \frac{9\pi}{10}$	154,57	144,89	85,58	154,52	144,74	85,14

Для такого большого количества образцов полученные оценки эффективных модулей Юнга не отличаются от точных значений (вычисленным по точной плотности трещин) больше чем на 1% для всех рассмотренных ориентаций. Однако при наличии меньшего количества образцов точность предсказания падает.

На рис. 2 представлена относительная разность между полученными оценками и точными значениями модулей Юнга в зависимости от количества образцов. График построен для ориентации, соответствующей последней строке таблицы ($\varphi_1 = \frac{\pi}{6}$, $\varphi_2 = \frac{9\pi}{10}$). Пунктирная линия отображает погрешность восстановления для модуля E_1, точечная – E_2, сплошная линия – E_3.

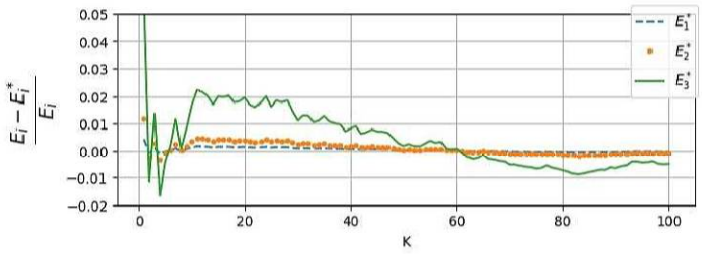


Рис. 2. Относительная разность между реальными и восстановленными модулями Юнга E_i в зависимости от количества рассматриваемых образцов K ($\varphi_1 = \frac{\pi}{6}$, $\varphi_2 = \frac{9\pi}{10}$, $r/\sqrt{S} = 0{,}1$, $N = 200$)

Можно видеть, что начиная с количества образцов $K = 40$ для количества трещин $N = 200$ относительная разность не превышает 1%. Самое большое относительное отклонение при любом количестве образцов соответствует наименьшему значению эффективного модуля Юнга (в данном случае он соответствует направлению вдоль оси Ox_3), самое маленькое – наибольшему (вдоль оси Ox_1). Кроме того, все три расчетных значения при фиксированном числе образцов одновременно либо завышены, либо занижены относительно точных значений.

5. Заключение. В работе предложен алгоритм оценки эффективных свойств материалов при наведенной анизотропии, вызванной появлением в материале системы параллельных трещин, по их видимым следам на поверхности образца. Проанализировано влияние количества образцов на расчетные значения модуля Юнга.

Литература

1. Kachanov M., Sevostianov I. Micromechanics of Materials, with Applications. Cham: Springer International Publishing AG part of Springer Nature, 2018. 712 p.

2. Абакаров А. М., Пронина Ю. Г. Об эффективных упругих свойствах материала со взаимно перпендикулярными системами параллельных трещин // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2022. Т. 18. № 1. С. 111–119.

3. Bristow J. R. Microcracks, and the static and dynamic elastic constants of annealed heavily cold-worked metals // British J. Appl. Phy. 1960. No 11. P. 81–85.

4. Pronina Y., Kachanov M. Estimating concentrations of cracks and platelets from their traces in 2D cross-sections // Mechanics of Materials. 2023. Vol. 180. Art. no 104618.

5. Вяльцева В. В., Пронина Ю. Г. Об оценке трехмерной плотности параллельных трещин по их следам на поверхности // Процессы управления и устойчивость. 2024. Т. 11. № 1. С. 88–92.

519.688

Гумеров И. И., Холодова С. Е.

Разработка методов математического моделирования динамических процессов во вращающейся электропроводящей жидкой среде

1. Введение. Мировой океан оказывает значительное влияние на жизнь человечества, делая изучение динамики морской среды ключевой задачей океанологических исследований. В статье [1] изучено влияние нелинейности ускорения вертикальных смещений упругой пластинки на дисперсионные характеристики волновых возмущений. Важную роль играет понимание поведения электропроводящих жидкостей в условиях вращения Земли и воздействия внешнего магнитного поля. В таких средах проявляются магнитогидродинамические эффекты – например, индукция магнитного поля, возникающая под действием течений и волн. Существующие подходы, такие как кинематические модели динамо [2], ограничены невозможностью учета полной системы магнитогидродинамических уравнений и диссипативных эффектов. В статье [3] изучалась роль стратификации в основной динамике изучаемых магнитогидродинамических процессов. Анализ полученного решения позволил установить факт существования установившегося режима колебаний при больших значениях времени для вмороженного в жидкую среду магнитного поля. Особый интерес представляют возмущения магнитогидродинамических величин с учетом диссипативных свойств жидкости, чему и посвящены исследования, представленные в данной статье.

2. Постановка задачи. Рассмотрим динамику электропроводящей жидкости, находящейся под воздействием вращения и внешнего магнитного поля. Для описания поведения такой жидкости используется система уравнений магнитогидродинамики, включающая уравнения движения и уравнения индукции магнитного поля. Пусть $\mathbf{v}$ – вектор скорости жидкости, $\mathbf{b}$ – вектор магнитной индукции, p –

Гумеров Ильсур Ильнурович – магистрант, университет ИТМО; e-mail: gumerovilsur618@gmail.com, тел.: +7(963)237-21-96

Холодова Светлана Евгеньевна – доцент, университет ИТМО; e-mail: kholodovase@itmo.ru, тел.: +7(965)021-00-41

давление, ρ – плотность жидкости, ω – угловая скорость вращения, μ – магнитная проницаемость, σ – электропроводность среды, $\mathbf{g}$ – ускорение силы тяжести. Тогда система уравнений магнитной гидродинамики имеет вид [4]

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v}, \nabla)\mathbf{v} = -\frac{\nabla p}{\rho} - 2[\omega, \mathbf{v}] + \mathbf{g}z + \frac{1}{\mu\rho}[\operatorname{rot}\mathbf{b}, \mathbf{b}],$$
$$\frac{\partial \mathbf{b}}{\partial t} = \operatorname{rot}[\mathbf{b} \times \mathbf{v}] + \frac{1}{\mu\sigma}\Delta\mathbf{b}, \quad \operatorname{div}\mathbf{b} = 0, \quad \operatorname{div}\mathbf{v} = 0,$$

с граничными условиями

$$v_z(x, y, -Z, t) = -v_x\frac{\partial Z}{\partial x} - v_y\frac{\partial Z}{\partial y}, \quad p(x, y, -h_B, t) = p_0(x, y, t),$$

$$v_z(x, y, -h_B, t) = -\frac{\partial h_B}{\partial t} - v_x\frac{\partial h_B}{\partial x} - v_y\frac{\partial h_B}{\partial y}, \quad b(x, y, -h_B, t) = b_0(x, y, t),$$

$$b_x\frac{\partial Z}{\partial x} + b_y\frac{\partial Z}{\partial y} + b_z = b_{n0}^{(e)}(x, y, t), \quad b_n(x, y, -h_B, t) = B_{n0}(x, y, t),$$

где $p_0, b_0, b_{n0}^{(e)}, B_{no}$ – заданные функции, $Z = Z(x, y)$ – известная поверхность дна, $h_B = h_B(x, y, t)$ – неизвестная свободная поверхность жидкости.

3. Решение редукционного уравнения и его анализ. В приближении длинных волн малой амплитуды решение исходной краевой задачи сводится к решению уравнения w

$$\mathcal{D}\left(\mathcal{D}_t^2 + \alpha^2\right)^2 \left\{ \left[\left(\mathcal{D}_t - \frac{\Delta}{R_m}\right)\mathcal{D}_t - \frac{\mathcal{D}^2}{\mu\rho}\right][\Delta_2\xi + \langle\nabla \ln H_0, \nabla\xi\rangle] - \right.$$
$$\left. - \frac{D\left(\ln H_0, \alpha\left(\mathcal{D}_t - \frac{\Delta}{R_m}\right)\xi\right)}{D(x, y)}\right\} +$$
$$+ \frac{1}{g(\mu\rho)^2 H_0}\left(\mathcal{D}_t^2 + \alpha^2\right)\left[F^2 + (\alpha\mathcal{D})^2\right]\left[\mathcal{D}\xi + \frac{B_{n0}}{2H_0\sqrt{f}}\langle\nabla\xi, \nabla(H_0 + Z)\rangle\right] =$$
$$= -\frac{\mathcal{D}H_0 + b_n^{(e)}}{(\mu\rho)^2 H_0} - \mathcal{D}\left(\mathcal{D}_t^2 + \alpha^2\right)^2\left[\left(\mathcal{D}_t - \frac{\Delta}{R_m}\right)\mathcal{D}_t - \frac{\mathcal{D}^2}{\mu\rho}\right] \times$$
$$\times [\Delta_2 P - \langle\nabla \ln H_0, \nabla P\rangle] + \alpha\mathcal{D}\left(\mathcal{D}_t^2 + \alpha^2\right)^2\left(\mathcal{D}_t - \frac{\Delta}{R_m}\right)\frac{D(\ln H_0, P)}{D(x, y)},$$

где $\eta(x,y,t) = h_B(x,y,t) - Z(x,y) - H_0(x,y)$ – функция возмущения свободной поверхности жидкости, $\xi(x,y,t)$ – некоторая модифицированная функция возмущения свободной поверхности жидкости, R_m – магнитное число Рейнольдса, $D = q_x\frac{\partial}{\partial x} + q_y\frac{\partial}{\partial y}, D_t = \frac{\partial}{\partial t}$, $F = \mu\rho(D_t^2 + \alpha^2)(D_t - \frac{\Delta}{R_m}) - D^2 D_t$ – введенные операторы [5].

Будем рассматривать решение уравнения в виде гармонической волны следующего вида:

$$\xi(x,y,t) = \operatorname{Re} A e^{i(kx+ly-\sigma t)},$$

где частота волновых возмущений σ является решением дисперсионного соотношения:

$$\begin{aligned}
&-(q_x k + q_y l)(\sigma^2 - \alpha^2)^2 \left[-\sigma^2 + \frac{(q_x k + q_y l)^2}{\mu\rho} - i\sigma\frac{(k^2+l^2)}{R_m}\right](k^2+l^2) + \\
&+ \frac{1}{g(\mu\rho)^2 H_0}(\alpha^2-\sigma^2)^2\left[(\mu\rho)^2(\alpha^2-\sigma^2)\left(-i\sigma + \frac{(k^2+l^2)}{R_m}\right)^2 + \right. \\
&\left. + (q_x k + q_y l)^4 + 2i\mu\rho\sigma\left(-i\sigma + \frac{(k^2+l^2)}{R_m}\right)(q_x k + q_y l)^2\right] \times \\
&\times (q_x k + q_y l) + \frac{B}{A(\mu\rho)^2 H_0} i = 0, \qquad (1)
\end{aligned}$$

где B – амплитуда гармоники разности индукции магнитного поля на границах.

Пусть далее уравнение поверхности, ограничивающей жидкий слой сверху, – $z = Z(x,y)$, где $Z(x,y) = Z$, Z – некоторая константа. Тогда невозмущенная глубина жидкого слоя также принимает некоторое постоянное значение H_0.

Решение дисперсионного уравнения реализовано численно на языке программирования C++. Основные шаги алгоритма:

1. На начальном этапе проверяется наличие тривиальных решений. Если такие корни существуют, они исключаются из дальнейшего рассмотрения.
2. Итерации Ньютона с модификациями [6]. Итерации прекращаются при достижении погрешности, которая определяется с учетом точности машинной арифметики.

3. Дефляция полинома. После нахождения корня полином делится на $x - x_0$, где x_0 – найденный корень, что снижает его степень и упрощает поиск оставшихся корней. Для устойчивости используется обратная дефляция (деление от младших коэффициентов).
4. Шаги 2–3 повторяются до тех пор, пока степень полинома не достигнет 2, оставшиеся два корня находятся через дискриминант.

Из уравнения (1) видно, что знак мнимой части σ определяет динамику гармоник. Пусть $\sigma = \sigma_1 + i\sigma_2$, тогда при $\sigma_2 < 0$ гармоника затухает, при $\sigma_2 > 0$ наблюдается неустойчивый режим, а при $\sigma_2 = 0$ – периодический режим.

В дальнейших расчетах все физические параметры, кроме тех, относительно которых рассматривается зависимость, принимаются равными 1, значение $g = 9{,}8$. На рис. 1 представлена зависимость σ_2 от магнитного числа Рейнольдса. Видно, что при равенстве амплитуд внешнего магнитного поля на ограничивающих жидкий слой поверхностях наблюдается существование трех ветвей дисперсионных кривых, одна из которых приводит к периодическому режиму изучаемого процесса, в то время как при разных амплитудах – восемь ветвей, ни одна из которых не приводит к периодическому режиму.

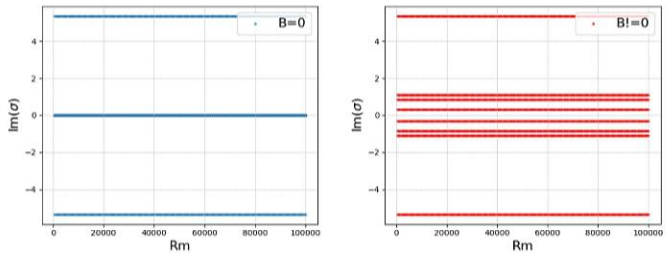


Рис. 1. График зависимости σ_2 от R_m

Зависимость мнимой части частоты волновых возмущений от удвоенной угловой скорости вращения жидкости приведена на рис. 2. При нулевой скорости вращения слоя и при равенстве амплитуд внешнего магнитного поля на границах наблюдается существование периодического режима. Совместное действие вращения и магнитного поля не влечет изменений в динамику рассматриваемого процесса

при различных условиях задания магнитного поля на ограничивающих жидкий слой поверхностях.

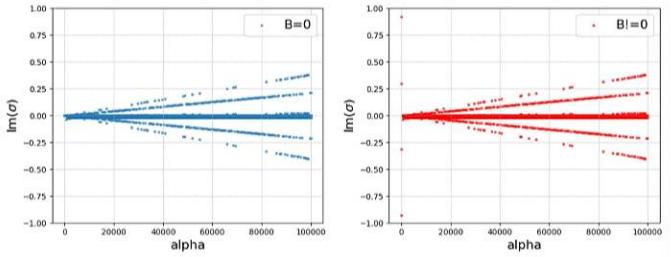


Рис. 2. График зависимости σ_2 от α

На рис. 3 представлена зависимость σ_2 от волнового числа. При всех значениях длины волны процесс будет являться неустойчивым, независимо от того, равны амплитуды внешнего магнитного поля на границах или нет, причем при возрастании волнового числа гармоника начинает расти с большей скоростью.

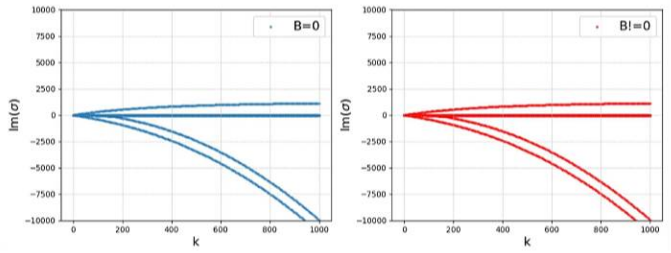


Рис. 3. График зависимости σ_2 от k

Приведем полученные аналитические представления для магнитогидродинамических характеристик исходной системы:

$$
\begin{aligned}
b_x(x,y,t) &= A\alpha^2\mu\rho(q_xk+q_yl)^3e^{\sigma_2 t}\Big(-2k\sigma_1\sigma_2\sin(kx+ly-\sigma_1 t)-\\
&\quad -k(\alpha^2-\sigma_1^2+\sigma_2^2)\cos(kx+ly-\sigma_1 t)\Big),\\
b_y(x,y,t) &= A\alpha^2\mu\rho(q_xk+q_yl)^3e^{\sigma_2 t}\Big(-2l\sigma_1\sigma_2\sin(kx+ly-\sigma_1 t)-\\
&\quad -l(\alpha^2-\sigma_1^2+\sigma_2^2)\cos(kx+ly-\sigma_1 t)\Big),\\
v_x(x,y,t) &= A\alpha^2(q_xk+q_yl)^4e^{\sigma_2 t}\Big(-k\sigma_1\cos(kx+ly-\sigma_1 t)+\\
&\quad +(l\alpha+k\sigma_2)\sin(kx+ly-\sigma_1 t)\Big),\\
v_y(x,y,t) &= A\alpha^2(q_xk+q_yl)^4e^{\sigma_2 t}\Big(-l\sigma_1\cos(kx+ly-\sigma_1 t)-\\
&\quad -(k\alpha-l\sigma_2)\sin(kx+ly-\sigma_1 t)\Big),\\
\eta(x,y,t) &= \frac{A\alpha^2(q_xk+q_yl)^4}{g}e^{\sigma_2 t}\Big((\sigma_1^2-\sigma_2^2)\cos(kx+ly-\sigma_1 t)-\\
&\quad -2\sigma_1\sigma_2\sin(kx+ly-\sigma_1 t)\Big),\\
b_z(x,y,z,t) &= -(H_0+Z+\eta+z)\Big(\frac{\partial b_x}{\partial x}+\frac{\partial b_y}{\partial y}\Big)-\\
&\quad -b_x\frac{\partial\eta}{\partial x}-b_y\frac{\partial\eta}{\partial y}+B_{n0}\sqrt{1+\Big(\frac{\partial\eta}{\partial x}\Big)^2+\Big(\frac{\partial\eta}{\partial y}\Big)^2},\\
v_z(x,y,z,t) &= -A\alpha^2(Z+z)(k^2+l^2)(q_xk+q_yl)^4e^{\sigma_2 t}\times\\
&\quad \times(\sigma_2\cos(kx+ly-\sigma_1 t)+\sigma_1\sin(kx+ly-\sigma_1 t)).
\end{aligned}
$$

Представленные аналитические выражения магнитогидродинамических характеристик позволяют сделать вывод о качественной динамике волнового режима в жидком вращающемся слое, находящемся во внешнем магнитном поле.

4. Заключение. Реализован метод поиска корней дисперсионного уравнения в комплексной области. Проведен анализ динамики вращающейся электропроводящей жидкости в приближении длинных волн малой амплитуды. Представленное исследование доказывает, что диффузия магнитного поля не всегда способствует его затуханию.

Литература

1. Букатов А. А. Анализ фазовой структуры колебаний жидкости с плавающей продольно сжатой упругой пластинкой при нелинейном взаимодействии поверхностных прогрессивных волн // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2020. Т. 16. № 3. С. 226–237.

2. Андреасян Р. Р., Михайлов Е. А., Андреасян А. Р. Структура и особенности формирования инверсий галактического магнитного поля // Астрономический журнал. 2020. Т. 97. № 3. С. 179–189.

3. Холодова С. Е. Математическое моделирование крупномасштабных движений стратифицированной электропроводной жидкости в сферическом слое // Вестник Санкт-Петербургского государственного университета. Прикладная математика. Информатика. Процессы управления. 2009. № 1. С. 117–132.

4. Peregudin S. I. Peregudina E. S., Kholodova S. E. On integration of the system of MHD equations modeling wave processes in a rotating liquid with arbitrary magnetic Reynolds number // Journal of Physics: Conference Series. 2020. No 1268. P. 257–272.

5. Kholodova S. E. Dynamics of a rotating layer of an ideal electrically conducting incompressible fluid // Zh. Vychisl. Mat. Mat. Fiz. 2008. Vol. 48. No 5. P. 882–898.

6. Madsen K. A. A root-finding algorithm based on Newton's method // BIT Numerical Mathematics. 1973. Vol. 13. P. 71–75.

УДК 004.942

Кикенов А. А.

Моделирование управляемого движения конвертоплана-трикоптера

Рекомендовано к публикации доцентом Шиманчуком Д. В.

1. Введение. Сейчас в технике как никогда актуальны вопросы управления [1, 2]. За последнее десятилетие развитие беспилотных летательных аппаратов (БПЛА) набрало новые обороты [3, 4]. Особый интерес представляют летательные аппараты (ЛА) вертикального взлета и посадки [5, 6]. Часто они представляют собой простое объединение мультикоптера и самолета что приводит к большому сопротивлению, создаваемому штангами в крейсерском режиме. Для решения этой проблемы можно использовать конвертопланы с изменяемым вектором тяги. Трехвинтовая схема позволит уменьшить вес ЛА, увеличить его управляемость и массу полезной нагрузки. Здесь исследуется математическая модель такого БПЛА, а также строится закон управления им.

2. Математическая модель. В работе [7] представлена полная математическая модель одного конвертоплана-трикоптера. Рассмотрим трикоптер той же конфигурации без учета выкоса заднего винта. Тогда кинематические уравнения останутся прежними. Динамические уравнения центра масс

$$m(\dot{\boldsymbol{v}} + \boldsymbol{\omega} \times \boldsymbol{v}) = \begin{bmatrix} F_x - mg\sin\theta + P_1\sin\varepsilon_1 + P_2\sin\varepsilon_2 \\ F_y - mg\cos\theta\cos\gamma + P_1\cos\varepsilon_1 + P_2\cos\varepsilon_2 + P_3 \\ F_z + mg\cos\theta\sin\gamma \end{bmatrix},$$

где $P_1 = Q\Omega_1^2$, $P_2 = Q\Omega_2^2$, $P_3 = q\Omega_3^2$ – силы тяг двигателей, зависящие от угловых скоростей вращения винтов; ε_1, ε_2 – углы поворота мотогондол, при этом при $\varepsilon_{1,2} = 0$ мотогондола установлена вертикально, $\varepsilon_{1,2} = \pi/2$ – горизонтально; $F_{x,y,z}$ – аэродинамические силы. Динамические уравнения вращательного движения принимают вид

$$I_s\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times I_s\boldsymbol{\omega} = v^2 C_m(\alpha, \beta, \boldsymbol{\delta}) +$$

Кикенов Арманжан Александрович – магистрант, Санкт-Петербургский государственный университет; e-mail: a.kikenov@spbu.ru тел.: +7(981)892-85-14

$$+\begin{bmatrix}\sin\varepsilon_1\\ \cos\varepsilon_1\\ 0\end{bmatrix}\left[\lambda Q\Omega_1^2 - I_2^p\dot\Omega_1\right]-\begin{bmatrix}\cos\varepsilon_1\\ -\sin\varepsilon_1\\ 0\end{bmatrix}\left[z_0 Q\Omega_1^2 + I_2^p\Omega_1(\dot\varepsilon_1-\omega_z)\right]+$$

$$+\begin{bmatrix}\sin\varepsilon_2\\ \cos\varepsilon_2\\ 0\end{bmatrix}\left[\lambda Q\Omega_2^2 - I_2^p\dot\Omega_2\right]+\begin{bmatrix}\cos\varepsilon_2\\ -\sin\varepsilon_2\\ 0\end{bmatrix}\left[z_0 Q\Omega_2^2 - I_2^p\Omega_2(\dot\varepsilon_2-\omega_z)\right]+$$

$$+\begin{pmatrix}0\\0\\1\end{pmatrix}\left[\cos\varepsilon_1(x_0 Q\Omega_1^2 - I_2^p\omega_x\Omega_1)-\sin\varepsilon_2(y_0 Q\Omega_1^2 - I_2^p\omega_y\Omega_1)\right]+$$

$$+\begin{pmatrix}0\\0\\1\end{pmatrix}\left[\cos\varepsilon_1(x_0 Q\Omega_2^2 - I_2^p\omega_x\Omega_2)-\sin\varepsilon_2(y_0 Q\Omega_2^2 - I_2^p\omega_y\Omega_2)\right]+$$

$$\begin{pmatrix}z_3\\ b\\ x_0\end{pmatrix}q\Omega_3^2-\dot\Omega_3\begin{pmatrix}0\\ I_2^d\\ 0\end{pmatrix}+\Omega_3\begin{pmatrix}\omega_z\\ 0\\ \omega_x\end{pmatrix}+I_3^o\begin{pmatrix}\omega_y(\dot\varepsilon_1+\dot\varepsilon_2)\\ \omega_x(\dot\varepsilon_1+\dot\varepsilon_2)\\ \ddot\varepsilon_1+\ddot\varepsilon_2\end{pmatrix},$$

где λ равен отношению геометрического момента винта к его силе тяги, а тензоры инерции остаются такими же, как и в статье [7].

Пусть $\Omega_{1,2,3}^2 \gg \Omega_{1,2,3} \gg 1$, $|\omega| \ll 1$, величины $\dot\varepsilon_{1,2,3}$ и $\ddot\varepsilon_{1,2}$ ограничены характеристиками сервопривода, а $I_3^o \ll I_z$. С помощью выбора управления можно ограничить величины производных $\Omega_{1,2,3}^2 \gg \dot\Omega_{1,2,3}$. Тогда часть слагаемых можно отбросить

$$\begin{pmatrix} m(\dot{\boldsymbol v}+\boldsymbol\omega\times\boldsymbol v)\\ I_s\dot{\boldsymbol\omega}+\boldsymbol\omega\times I_s\boldsymbol\omega\end{pmatrix} = B\boldsymbol\eta + v^2\boldsymbol c_m(\alpha,\beta,\boldsymbol\delta)+\boldsymbol f(\theta,\gamma),$$

$$B=\begin{pmatrix}0&1&0&1&0\\1&0&1&0&1\\0&0&0&0&0\\-z_0&\lambda&z_0&-\lambda&z_3\\\lambda&z_0&-\lambda&-z_0&b\\x_0&-y_0&x_0&-y_0&x_3\end{pmatrix},\quad f(\theta,\gamma)=mg\begin{bmatrix}-\sin\theta\\-\cos\theta\cos\gamma\\\cos\theta\sin\gamma\\0\\0\\0\end{bmatrix},$$

$$\boldsymbol\eta=[P_1\cos\varepsilon_1, P_1\sin\varepsilon_1, P_2\cos\varepsilon_2, P_2\sin\varepsilon_2, P_3]^T,$$

$\boldsymbol{c_m}$ – вектор аэродинамических коэффициентов, α и β – углы атаки и скольжения, $\boldsymbol\delta$ – вектор управлений рулевыми плоскостями.

Пусть $\boldsymbol u = B\boldsymbol\eta$. Матрица B является невырожденной, что позволяет получать управление $\boldsymbol\eta = B^{-1}\boldsymbol u$.

3. Построение системы управления и моделирование. Пусть управление $\boldsymbol{u} = B\boldsymbol{\eta}$ осуществляется регулятором, схожим по структуре с регулятором из статьи [4]. Здесь можно управлять одновременно пятью величинами: v_x, v_y, Ω_x, Ω_y, Ω_z.

3.1. Режим стабилизации положения равновесия. Если положить $P_1 \cos\varepsilon_1 + P_2 \cos\varepsilon_2 \equiv 0$, то структурно получим управление по-мультикоптерному, когда положение регулируется с помощью углов тангажа и крена (рис. 1).

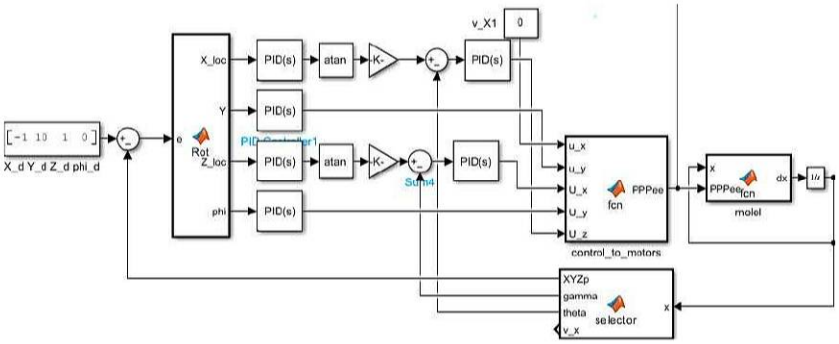


Рис. 1. Структура управления по-коптерному

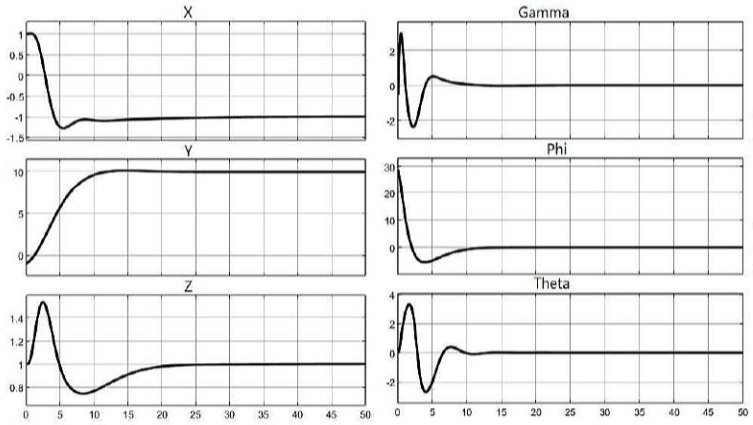


Рис. 2. Стабилизация в коптерном режиме

Пренебрегая сопротивлением среды, стабилизируем вектор $[X, Y, Z, \varphi]$ с помощью регулятора, представленного на рис. 1. Результаты представлены на рис. 2.

Используя канал управления по v_x, т. е. $P_1 \cos \varepsilon_1 + P_2 \cos \varepsilon_2 \not\equiv 0$ (рис. 3), можно построить иной регулятор, при котором угол $\theta \equiv 0$ не меняется для стабилизации по X.

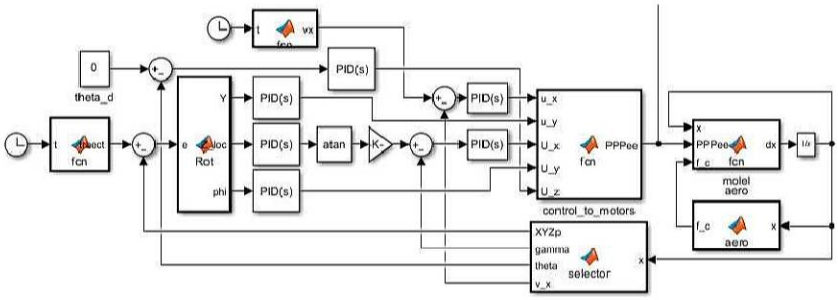


Рис. 3. Структура управления по-конвертопланному

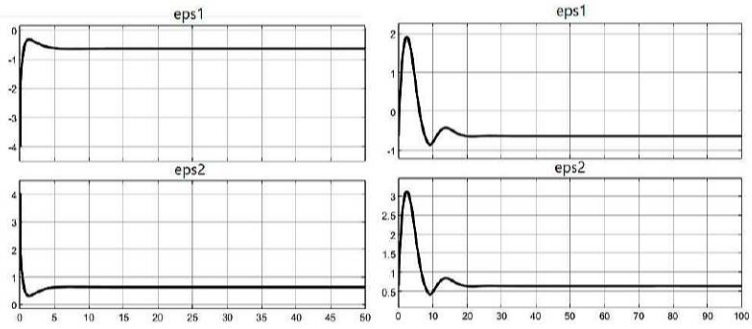


Рис. 4. Управление $\varepsilon_{1,2}$ в коптерном режиме (слева) и в конвертопланном режиме (справа)

Рис. 4 иллюстрирует разницу в управлении мотогондолами в режиме стабилизации между двумя регуляторами.

3.2. Режим стабилизации динамического положения равновесия. Используя пятый канал управления по v_x (рис. 3), мы также можем снизить расход топлива при горизонтальном движении, так как в этом случае есть возможность двигаться вперед с углом атаки, дающим минимум сопротивления воздуха.

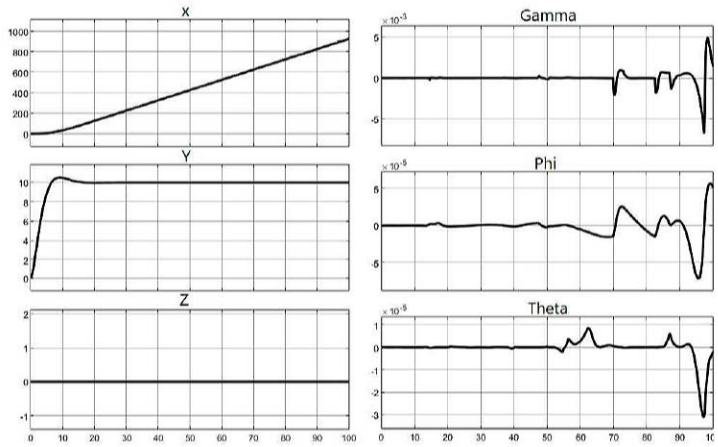


Рис. 5. Фазовые переменные в конвертопланном режиме

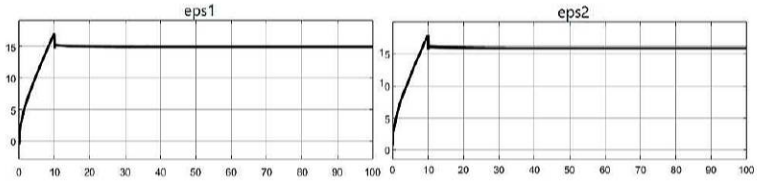


Рис. 6. Положение мотогондол при горизонтальном полете по-конвертопланному

Далее будем моделировать горизонтальное движение ЛА около динамического положения равновесия $v_x = \min(t, 10)$, $Z = 0$, $Y = 10$, $\varphi = 0$, $\theta = 0$, $t \in [0, 100]$. Результаты представлены на рис. 5. Изменение углов наклона винтов показано на рис. 6.

4. Заключение. Построена математическая модель конвертоплана, синтезировано два PID-регулятора, а также промоделировано управляемое движение конвертоплана около статического и динамического положений равновесия.

Литература

1. Смирнова М. А., Смирнов М. Н., Смирнов Н. В. Система многоцелевого управления роботом-манипулятором // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2022. Т. 18. № 4. С. 621–630.

2. Жабко А. П., Жабко Н. А., Яковлев П. В. Метод оптимального демпфирования В. И. Зубова в задаче управления одной гироскопической системой // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2022. Т. 18. № 2. С. 278–284.

3. Баранов О. В., Жолобов Е. В. Моделирование аварийных управляемых движений квадрокоптера // Процессы управления и устойчивость. 2019. Т. 6. № 1. С. 213–217.

4. Нечипорук А. А., Лепихин Т. А., Щербинин А. В. Моделирование и управление квадрокоптером // Процессы управления и устойчивость. 2017. Т. 4. № 1. С. 445–451.

5. Черноморский А. И., Лельков К. С., Курис Э. Д. и др. Структура и законы управления тягой четырехмоторного конвертоплана в переходном режиме его полета // Известия Тульского государственного университета. Технические науки. 2021. С. 37–44.

6. Houari A., Bachir I., Mohamed D. K., Kara-Mohamed M. PID vs LQR controller for tilt rotor airplane // International Journal of Electrical and Computer Engeneering. 2020. Vol. 10. No 6. P. 6309–6318.

7. Кикенов А. А. Математическая модель конвертоплана // Процессы управления и устойчивость. 2024. Т. 11. № 1. С. 107–111.

УДК 621.384.639

Князев Н. А., Левинский Д. В.

Учет изменений магнитных полей диполей при моделировании поперечного движения частиц в синхротроне

Рекомендовано к публикации профессором Овсянниковым Д. А.

1. Введение. Крупнейшие исследовательские установки класса «Мегасайенс» применяются в различных областях науки, включая физику, химию, биологию, медицину и т. д. Среди данных установок особое значение имеют источники синхротронного излучения (ИСИ), в которых дипольные магниты являются одними из ключевых элементов. Однако проектирование таких магнитов сопряжено с необходимостью балансировки между высокими требованиями к точности магнитного поля, инженерными ограничениями и экономической целесообразностью. Чтобы минимизировать риски несоответствия реальных характеристик магнитов их теоретическим моделям, необходимо на ранних стадиях проектирования проводить анализ и оценку конструкций путем моделирования оптической системы.

2. Постановка задачи. Цель исследования заключается в разработке и настройке параметров модели короткого дипольного магнита, построенной на основе расчетного пространственного распределения магнитного поля (рис. 1) [1–3], соответствующего физическим требованиям конфигурации элемента (шунтирование, постоянные высококоэрцитивные магниты). В качестве инструмента для выполнения оптических расчетов использовался комплекс программ MAD-X [4].

3. Описание методологии. Данный раздел посвящен описанию методологической базы программного комплекса MAD-X, включая технику матричного отображения и следящую систему координат, а также предложенной модели короткого магнита.

Князев Никита Андреевич – аспирант, Санкт-Петербургский государственный университет; e-mail: st064730@student.spbu.ru, тел.: +7(908)027-92-59

Левинский Дмитрий Вячеславович – студент, Санкт-Петербургский государственный университет; e-mail: st097344@student.spbu.ru, тел.: +7(953)502-02-33

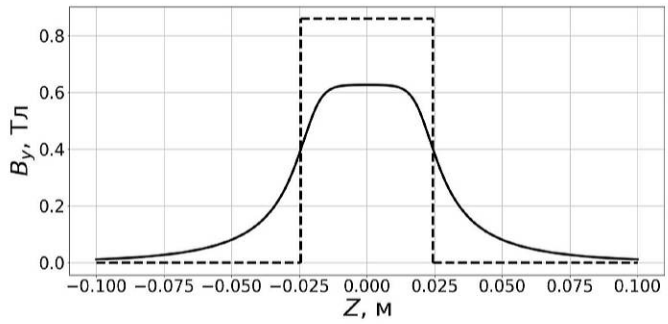


Рис. 1. Распределение вертикальной компоненты поля короткого магнита вдоль его оси. Пунктирная линия соответствует теоретическому распределению, сплошная – расчетному

3.1. Моделирование на основе транспортных матриц. Уравнения движений траекторий частиц в магнитных полях могут быть представлены в канонической форме, и оптический канал рассматривается в параксиальном приближении. Каждый элемент схемы представляется с помощью транспортной матрицы, а совокупность элементов (суперпериод или все кольцо) описывается как суперпозиция транспортных матриц, входящих в моделируемую систему. Преимуществами использования метода матричного отображения (транспортных матриц) являются простота, универсальность, скорость и точность.

Таким образом, зная начальные координаты частицы и матрицы преобразования для каждого элемента оптического канала, можно определить координаты частицы на выходе из канала путем последовательного умножения вектора начальных координат на транспортные матрицы соответствующих участков.

3.2. Следящая система координат. Следящая (подвижная) система координат – это система отсчета, которая динамически изменяет свою ориентацию и положение, следуя за траекторией частицы или ее усредненным движением. Это позволяет упростить анализ сложного движения частицы в ускорителе, разделяя его на компоненты:

- основное движение – описывается в глобальной системе координат;

- отклонения от траектории – описываются в следящей системе координат.

В циклических ускорителях следящая система координат обычно привязана к равновесной орбите. Для упрощения анализа движения частицы, движущейся по искривленной траектории в магнитном поле, используется трехгранник Френе – ортонормированный базис, связанный с эталонной частицей. Он состоит из трех векторов: касательного, нормального и бинормального, которые определяют мгновенное направление движения частицы и ее отклонения. Это позволяет записать уравнения движения в локальной системе координат, что дает возможность раздельно рассматривать продольную и поперечную динамику пучка.

3.3. Моделирование короткого магнита. Для учета формы поля короткого магнита и краевых эффектов предлагается заменить пролетные промежутки и сам диполь на совокупность достаточно большого числа расположенных последовательно «идеальных» магнитов с полями, соответствующими расчетному распределению поля. Таким образом, поле исследуемого короткого магнита моделируется с помощью кусочно-постоянного приближения (рис. 2).

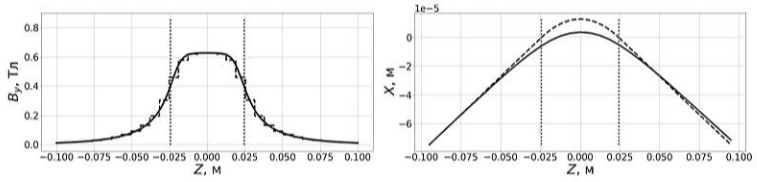


Рис. 2. Модель и отклонение траекторий для кусочно-постоянной аппроксимации короткого магнита с 30 разбиениями. Сплошная линия соответствует расчетному распределению, пунктирная – смоделированному

При замене идеального магнитного поля на кусочно-постоянную аппроксимацию, соответствующую заданной карте поля, изменится равновесная орбита. Для модели короткого диполя с разбиением на

30 частей оценка отклонения равновесной орбиты представлена на рис. 2. Так как следящая система координат при расчете траектории частицы в накопительном кольце связана с равновесной орбитой, то необходимо добавить поправки по координате и углу (импульсное воздействие), полученные в конце моделируемого элемента, чтобы учитывать изменения «новой» равновесной орбиты.

4. Вычислительные эксперименты. При моделировании короткого магнита описанным ранее методом основным параметром является количество идеальных диполей, которые используются для аппроксимации расчетного распределения поля. Под идеальным элементом будем считать элемент, соответствующий первоначальным теоретически заданным параметрам структуры. Чтобы определить оптимальное количество разбиений короткого диполя, выполним численный эксперимент. Построим равновесные траектории с различным числом дроблений – от 10 до 1000, используя расчетное распределение магнитного поля. Затем для каждой траектории определим отклонения в конечной точке магнита от равновесной орбиты для модели, где короткий диполь разделен на 1000 частей (эту конфигурацию считаем предельной). Результаты представлены на рис. 3.

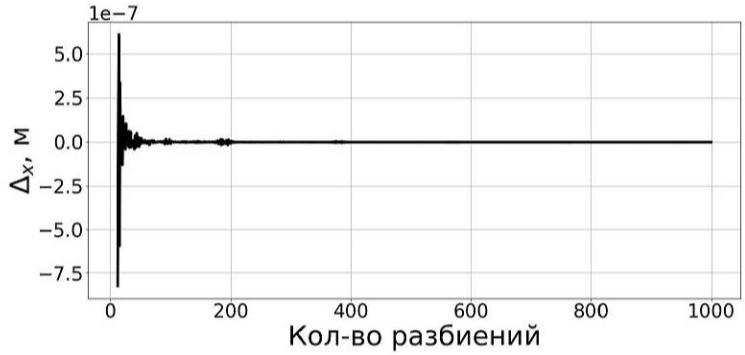


Рис. 3. Отклонения выходной координаты для моделей короткого магнита относительно модели с 1000 разбиений

Анализ полученных данных показывает, что при дроблении короткого магнита на 200 и более частей разница в отклонениях траекторий становится пренебрежимо малой. С учетом этого, для обеспечения надежности вычислений выберем количество разбиений равным 250. Далее построим фазовые эллипсы для модели кусочно-постоянной аппроксимации короткого магнита с 250 разбиений и сопоставим полученные результаты с идеальной моделью. Результаты выполненных расчетов представлены на рис. 4.

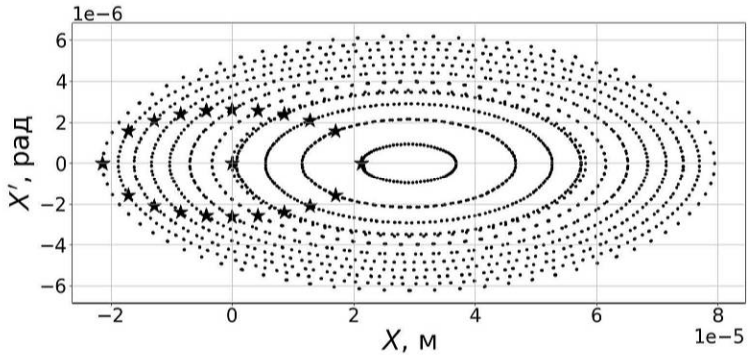


Рис. 4. Фазовые эллипсы для идеальной модели (звездочка) и модели кусочно-постоянной аппроксимации (точка) короткого магнита с 250 разбиений на 100 полных оборотах

Для модели кусочно-постоянной аппроксимации короткого магнита с 250 разбиений фазовый эллипс пучка располагается со смещением в положительную сторону относительно идеальной равновесной орбиты. При этом небольшие отклонения равновесной орбиты можно компенсировать за счет корректоров или изменения начального положения инжектируемого пучка.

5. Заключение. В работе составлены модели короткого диполя накопительного кольца синхротрона, основанные на расчетном распределении магнитного поля. Также проведены анализ и оценка полученных моделей. Вычислительные эксперименты показали, что оптимальное число разбиений короткого магнита для модели кусочно-постоянного приближения соответствует 250. На основании полученных результатов можно сделать вывод, что замена короткого магнита с моделью постоянного поля на модель кусочно-постоянной аппроксимации не приводит к существенным изменениям характеристик пучка заряженных частиц.

Результаты исследования могут быть использованы при разработке и проектировании дипольных магнитов синхротрона, в частности, УНК-СИЛА.

6. Благодарности. Авторы признательны за поддержку, дискуссии и плодотворные обсуждения Д. А. Овсянникову, А. Д. Овсянникову, В. А. Козынченко, Е. Д. Котиной (СПбГУ), Н. Ю. Казаринову, И. А. Иваненко (ОИЯИ ЛЯР СФУ), Н. И. Казаченко, В. П. Кухтину, Е. А. Ламзину, С. Е. Сычевскому (АО «НИИЭФА»).

Литература

1. Arslanova D., Gavrish Y. et al. Feasibility study of permanent magnet dipoles for SILA facility // Cybernetics and physics. 2023. No 12. P. 252–256.

2. Беляева А. А., Князев Н. А. Влияние модификаций вычислительных моделей постоянных дипольных магнитов источника синхротронного излучения на оптические характеристики пучка заряженных частиц // Процессы управления и устойчивость. 2024. Т. 11. № 1. P. 71–80.

3. Синхротронный комплекс СИЛА // Научная Россия [Электронный ресурс]: URL: https://scientificrussia.ru/partners/rossijskaya-akademiya-nauk/sila-rossijskoj-nauki (дата

4. MAD – Methodical Accelerator Design // CERN [Электронный ресурс]: URL: https://madx.web.cern.ch/ (дата обращения: 18.03.2025). обращения: 15.03.2025).

УДК 517.958, 539.3

Муллаянов Т. О.

Влияние формы наноразмерного рельефа на распределение напряжений вблизи свободной поверхности твердого тела

Рекомендовано к публикации доцентом Костырко С. А.

1. Введение. В последнее время растет интерес к изучению свойств наноразмерных структур [1–3]. Создание таких структур с определенными функциональными характеристиками является одним из приоритетных направлений современной микроэлектроники и оптики. Комбинирование разнообразных материалов и структурных элементов на наноскопическом уровне позволяет создавать устройства с уникальными электрическими, оптическими и магнитными характеристиками. Но при производстве таких материалов, а также в процессе их термической или химической обработки могут возникать поверхностные дефекты в виде канавок и трещинообразных впадин [4]. В ходе эксплуатации эти дефекты могут эволюционировать, что приводит к деградации свойств материалов и сокращению срока службы приборов, в которых они используются. Исследования показывают, что плоская поверхность напряженного твердого тела при определенных условиях становится неустойчивой к малым синусоидальным возмущениям. Основным механизмом потери устойчивости в таких системах является поверхностная диффузия, которая приводит к изменению формы поверхности. Феномен волнообразования был подтвержден многочисленными экспериментальными исследованиями [5, 6]. Наряду с волнами синусоидальной формы возможно образование острых выступов и трещинообразных впадин [7]. Для точного прогнозирования разрушения наноразмерных структур необходимо с высокой точностью определять напряженное состояние вблизи поверхностных топологических дефектов.

В данном исследовании для описания формы дефектов применяется представление в виде рядов Фурье, что позволяет эффективно

Муллаянов Тимур Олегович – магистрант, Санкт-Петербургский государственный университет; e-mail: st087024@student.spbu.ru, тел.: +7(987)053-38-01

моделировать поверхности сложной формы. Для определения напряженного состояния вблизи искривленной поверхности наноматериала используется модель Гертина – Мердока, в которой приповерхностный слой твердого тела представляется в виде упругой мембраны, идеально связанной с объемной фазой. Решение сформулированной краевой задачи строится на основе метода возмущения, где в качестве малого параметра выбрано отношение амплитуды к длине волны рельфа. В рамках проведенных ранее исследований [3] показано, что для синусоидального рельефа линейное приближение метода возмущений недостаточно точно прогнозирует напряжения вблизи искривленной границы. Это связано с тем, что возмущенное состояние вблизи таких поверхностей включает нелинейные компоненты, не учитываемые в первом приближении. В данной работе рассматривается влияние остаточных напряжений и поверхностной упругости на концентрацию напряжений для более сложного рельефа, который образуется в результате действия поверхностной диффузии. Учтенные эффекты позволяют лучше понять механизмы формирования локальных областей повышенных напряжений и прогнозировать эволюцию поверхностных дефектов в процессе эксплуатации материалов.

2. Постановка задачи. Учитывая условия плоской деформации, материал с наноструктурированной поверхностью моделируется в виде полуплоскости Ω с искривленной границей Γ:

$$\Omega = \{z = x_1 + ix_2 : x_2 < \varepsilon f(x_1), x_1 \in \Re^1\},$$
$$\Gamma = \{\zeta = x_1 + ix_2 : x_2 = \varepsilon f(x_1), x_1 \in \Re^1\}, \quad 0 < \varepsilon \ll 1.$$

Напряженное состояние на границе Γ, вызванное действием поверхностных напряжений, принимает вид

$$\sigma_{nn}(\zeta) + i\sigma_{nt}(\zeta) = q^s(\zeta), \quad \zeta \in \Gamma, \tag{1}$$

где σ_{nn} и σ_{nt} – компоненты тензора напряжений в локальной системе координат n, t, связанной с искривленной поверхностью.

Дополнительное уравнение, определяющее неразрывность перемещений на поверхности, имеет следующий вид:

$$u^s(\zeta) = u(\zeta), \quad \zeta \in \Gamma,$$

где $u^s = u_1^s + iu_2^s$ и $u = u_1 + iu_2$ – перемещения поверхностной и объемной фазы вдоль осей x_1 и x_2 соответственно.

Согласно модели Гертина – Мердока [8, 9] функция q^s принимает вид [10]

$$q^s(\zeta) = \frac{\sigma_0^s}{R} + \left[\frac{M_s}{R}\mathrm{Re}\frac{\partial u}{\partial \zeta} + \sigma_0^s \mathrm{Im}\left(\frac{\partial^2 u}{\partial \zeta^2} e^{i\alpha_0}\right)\right] - \\ - \left[M_s \mathrm{Re}\left(\frac{\partial^2 u}{\partial \zeta^2} e^{i\alpha_0}\right) - \frac{\sigma_0^s}{R}\mathrm{Im}\frac{\partial u}{\partial \zeta}\right], \tag{2}$$

где R – радиус кривизны границы Γ; α_0 – угол между касательной к Γ и осью x_1 в точке ζ; σ_0^s – поверхностное напряжение; λ_s, μ_s – константы поверхностной упругости, аналогичные константам Ламе λ, μ объемной фазы наноматериала, $M_s = \lambda_s + 2\mu_s$. Условия (1), (2) описывают обобщенный закон Юнга – Лапласа [11].

Принимая во внимание, что

$$e^{i\alpha_0} = \frac{1 + i\varepsilon f'(x_1)}{h(x_1)}, \qquad h(x_1) = \sqrt{1 + \varepsilon^2 f'^2(x_1)},$$

выражение (2) перепишем в виде

$$q^s(\zeta) = \kappa\sigma_0^s + \frac{\kappa}{2}\left[p_s \frac{\partial u}{\partial \zeta} + m_s \overline{\frac{\partial u}{\partial \zeta}}\right] - \\ - \frac{1}{2h}\left[p_s(i - \varepsilon f')\frac{\partial^2 u}{\partial \zeta^2} + m_s(i + \varepsilon f')\overline{\frac{\partial^2 u}{\partial \zeta^2}}\right],$$

где $p_s = M_s + \sigma_0^s$, $\quad m_s = M_s - \sigma_0^s$, h – метрический коэффициент, κ – кривизна границы Γ [10, 12].

Для решения поставленной краевой задачи используется подход, разработанный в [10, 12] на основе использования метода комплексных потенциалов и метода граничных возмущений. Вектор напряжений $\sigma = \sigma_{nn} + i\sigma_{nt}$ выражается через комплексные потенциалы Гурса – Колосова Φ и Υ. Далее представляем функции $\Phi(z)$, $\Upsilon(z)$ и $u'(z)$ в виде степенных рядов по малому параметру ε.

В результате чего исходная краевая задача сводится к рекуррентной последовательности интегральных уравнений

$$2\mu u'_{(n)}(x_1) + \frac{\varkappa + 1}{2\pi i}\int\limits_{-\infty}^{+\infty} \frac{q^s_{(n)}(t)}{x_1 - 1}dt + \frac{1 - \varkappa}{2} q^s_{(n)}(x_1) = F_n(x_1) \tag{3}$$

относительно коэффициентов разложения комплексного вектора поверхностных перемещений в степенной ряд

$$u'_n(x_1) = \sum_{k=1}^{\infty} \left[C_k^n e^{ib_k x_1} + C_{-k}^n e^{-ib_k x_1} \right], \quad b_k = 2\pi k/a. \tag{4}$$

В выражении (3) $n = 1 \dots N$; $\varkappa = (\lambda + 3\mu)/(\lambda + \mu)$; $u'_{(n)}$ и $q^s_{(n)}$ – коэффициенты разложения в степенной ряд соответствующих функций; F_n – известные функции, зависящие от предыдущих аппроксимаций $u'_{(1)} \dots u'_{(n-1)}$. Для получения коэффициентов C_k^n и C_{-k}^n в выражении (3) правая часть представляется в виде ряда Фурье, в $u'_{(n)}$ подставляется (4), приравниваются соответствующие коэффициенты.

3. Численные результаты. Для численного моделирования используем упругие константы, соответствующие алюминию: поверхностные модули $\lambda_s = 6{,}851$ Н/м, $\mu_s = -0{,}376$ Н/м, что дает $M_s = 6{,}099$ Н/м; объемные упругие параметры $\lambda = 58{,}17$ ГПа, $\mu = 26{,}13$ ГПа.

Геометрия поверхности задается функцией

$$f(x, y) = \frac{a\,(\mathrm{Im}(\mathrm{ctg}(\pi x/a - iy)) - 1)}{\mathrm{Im}(\mathrm{ctg}(iy)) + 1},$$

где параметр y определяет форму рельефа поверхности [13].

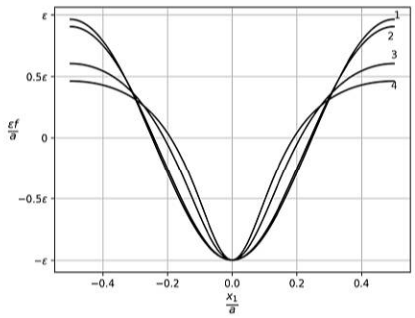


Рис. 1. Функция, описывающая рельеф [13]

Это позволяет рассматривать различные профили поверхности, которые наблюдаются в теоретических работах, посвященных изу-

чению морфологической потери устойчивости твердых тел на мезоскопическом уровне. На рис. 1 цифрами 1–4 отмечены значения $y = 2$; 1,5; 0,7; 0,5 соответственно. Далее показаны зависимости напряжений при различных формах рельефа с учетом второго порядка метода возмущений.

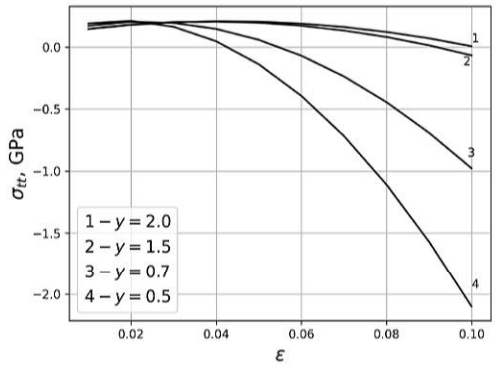


Рис. 2. Зависимость напряжений σ_{tt} от параметра ε для различных форм рельефа

Анализ полученных результатов (см. рис. 2) показал, что по мере роста малого параметра ε точность второго порядка приближения снижается, а при учете сложной формы рельефа становится очевидной необходимость использования более высоких порядков.

Литература

1. Шувалов Г. М. Влияние упругих свойств гетероэпитаксиальной системы на устойчивость границы соединения двух материалов // Процессы управления и устойчивость. 2020. Т. 7. № 1. С. 133–137.

2. Кондратьева А. Д. Выпучивание под собственным весом нанокантилевера с упругим поворотом // Процессы управления и устойчивость. 2024. Т. 11. № 1. С. 112–120.

3. Mullayanov T. O. Investigating elastic response of nanostructured solid surface using boundary perturbation method // Процессы управления и устойчивость. 2024. Т. 11. № 1. P. 134–138.

4. Freund L. B., Suresh S. Thin film materials: stress, defect formation and surface evolution. Cambridge: Cambridge university press, 2004. 770 p.

5. Yao J. Y., Andersson T. G., Dunlop G. L. Microstructures and critical thicknesses of InxGa1-xAs/GaAs strained-layer structures // Semiconductor science and technology. 1994. Vol. 9. No 5. P. 1086–1095.

6. Eaglesham D. J., Cerullo M. Dislocation-free stranski-krastanow growth of Ge on Si(100) // Physical review letters. 1990. Vol. 64. No 16. P. 1943–1946.

7. Jesson D. E. et al. Direct imaging of surface cusp evolution during strained-layer epitaxy and implications for strain relaxation // Physical review letters. 1993. Vol. 71. No 11. P. 1744–1747.

8. Gurtin M. E., Murdoch A. I. A continuum theory of elastic material surfaces // Archive for rational mechanics and analysis. 1975. Vol. 57. P. 291–323.

9. Gurtin M. E., Murdoch A. I. Surface stress in solids // International journal of Solids and Structures. 1978. Vol. 14. No 6. P. 431–440.

10. Grekov M. A. General approach to the modified Kirsch problem incorporating surface energy effects // Continuum Mechanics and Thermodynamics. 2021. Vol. 33. No 4. P. 1675–1689.

11. Duan H. L., Wang J., Karihaloo B. L. Theory of elasticity at the nanoscale // Advances in applied mechanics. 2009. Vol. 42. P. 1–68.

12. Kostyrko S., Grekov M., Altenbach H. Stress distribution at the wavy surface of a solid incorporating surface stresses and surface tension // Advances in Solid and Fracture Mechanics. Springer, Cham. 2022. P. 151–166.

13. Grekov M. A., Kostyrko S. A. Surface effects in an elastic solid with nanosized surface asperities // International Journal of Solids and Structures. 2016. Vol. 96. P. 153–161.

УДК 004.855.5

Ручьев И. А., Шиманчук Д. В.

Методы машинного обучения для решения обратной задачи кинематики робота-манипулятора с избыточными числом степеней свободы

1. Введение. Одной из ключевых и наиболее сложных проблем в робототехнике является решение обратной задачи кинематики, которая заключается в определении значений углов для каждого сустава многозвенного механизма, обеспечивающих заданное положение роботизированной системы в пространстве. Несмотря на наличие различных методов решения данной задачи, многие из них обладают существенными ограничениями, такими как высокая вычислительная сложность, ограниченность множества решений, а в случае избыточных механизмов – отсутствие гарантии нахождения оптимального решения.

Управление роботами-манипуляторами, при котором сервомоторы программируются в зависимости от положения инструмента, требует преобразования координат инструмента в обобщенные координаты манипулятора. Однако такой подход применим лишь для ограниченного класса роботов (например, FANUC Robotics M-20iA), что обусловлено конечностью множества решений обратной задачи кинематики. Основная сложность заключается в избыточности степеней свободы: более трех – для плоских и более шести – для пространственных манипуляторов. В общем случае кинематическое преобразование между пространством координат инструмента и пространством конфигураций манипулятора является нелинейным, а решение обратной задачи кинематики, выражаемое через вектор обобщенных координат, неединственно.

Эти особенности послужили основой для разработки различных методов управления роботами-манипуляторами с избыточным числом степеней свободы, где решение может быть найдено на основе

Ручьев Иван Анатольевич – студент, Санкт-Петербургский государственный университет; e-mail: st109450@student.spbu.ru, тел.: +7(965)098-04-78

Шиманчук Дмитрий Викторович – доцент, Санкт-Петербургский государственный университет; e-mail: d.shimanchuk@spbu.ru, тел.: +7(906)240-51-19

заданных критериев оптимизации и ограничений на обобщенные координаты.

В последние годы методы машинного обучения находят все более широкое применение в решении сложных задач благодаря их способности выявлять нелинейные зависимости при относительно низких вычислительных затратах. В данной работе предлагается подход к решению обратной задачи кинематики с использованием методов машинного обучения. Проведено сравнение эффективности нескольких алгоритмов: полносвязной нейронной сети, метода случайного леса и бустингового алгоритма на основе решающих деревьев.

2. Об обратной задаче кинематики в задачах управления. Робот-манипулятор представляет собой многозвенную механическую систему, соединенную шарнирами, где один конец цепи закреплен, а другой остается свободным и называется рабочим органом (или инструментом) манипулятора. Шарниры могут быть призматическими или вращательными и приводятся в движение сервомоторами. Для перемещения рабочего органа по заданной траектории необходимо согласованное изменение положений звеньев манипулятора. В каждый момент времени необходимо знать, как перемещаются звенья относительно фиксированной системы отсчета, которая называется базовой. В ней определяется траектория движения рабочего органа.

Кинематическая модель манипулятора описывает положение, скорость и ускорение звеньев без учета масс, сил и моментов. Прямая задача кинематики заключается в нахождении положения и ориентации рабочего органа $\mathbf{s}$ в базовой системе координат для заданного вектора обобщенных координат $\mathbf{q}$ (углов сочленений):

$$\mathbf{s} = f(\mathbf{q}), \tag{1}$$

где f – нелинейная, дифференцируемая функция. Уравнение (1) имеет единственное решение. Обратная задача кинематики связана с нахождением вектора обобщенных координат $\mathbf{q}$ для заданного положения и ориентации рабочего органа $\mathbf{s}$. Она сводится к решению уравнения

$$\mathbf{q} = f^{-1}(\mathbf{s}). \tag{2}$$

В отличие от прямой задачи, решение уравнения (2) в общем случае неединственно, что обусловлено возможностью существования

множества конфигураций манипулятора, обеспечивающих одинаковое положение рабочего органа.

Задача управления движением манипулятора заключается в перемещении рабочего органа в желаемое положение с заданной ориентацией, следуя определенной траектории в обобщенных координатах. Однако, если управление осуществляется в пространстве координат инструмента, то необходимо знать решение обратной задачи кинематики [1].

3. Постановка задачи. Обратная задача кинематики требует нахождения отображения

$$\mathbf{q} = f^{-1}(\mathbf{s}) \approx F_\omega(\mathbf{s}),$$

где F_ω – модель машинного обучения, аппроксимирующая обратное отображение, а ω – параметры модели, которые настраиваются в процессе обучения. Для оценки качества модели используется среднеквадратичная ошибка:

$$\mathrm{MSE}(\mathbf{s}, \mathbf{q}) = \frac{1}{N} \sum_{i=1}^{N} \left(\mathbf{q}_i - F_\omega(\mathbf{s}_i)\right)^2,$$

где N – размер выборки. В данной статье для решения задачи использовались три метода машинного обучения: случайный лес, бустинговый алгоритм XGBoost и полносвязная нейронная сеть. Каждый из этих методов был выбран благодаря способности эффективно работать с нелинейными зависимостями [2], что особенно важно для задачи обратной кинематики.

4. Случайный лес. Случайный лес – это ансамблевый метод, основанный на построении множества решающих деревьев. Методом бутстреппинга для заданного количества деревьев k формируется k случайных подвыборок с повторениями, на основе каждой из которых строится решающее дерево [3]. Итоговый прогноз модели формируется как среднее значение предсказаний всех деревьев:

$$F_\omega(\mathbf{s}) = \frac{1}{k} \sum_{i=1}^{k} h_i(\mathbf{s}),$$

где $h_i(\mathbf{s})$ – предсказание i-го дерева, а k – общее число деревьев.

Пространством параметров модели ω в этом случае выступают условия ветвления деревьев и значения на листьях.

5. Бустинговый алгоритм XGBoost. В отличие от случайного леса, XGBoost строит ансамбль деревьев последовательно, где каждое последующее дерево корректирует ошибки предыдущих. Остаточная ошибка для каждого нового дерева вычисляется как градиент функции потерь относительно предсказаний предыдущего ансамбля. Для MSE функция потерь на одном примере имеет вид

$$L(\mathbf{q}, F_\omega(\mathbf{s})) = \frac{1}{2}(\mathbf{q} - F_\omega(\mathbf{s}))^2.$$

Градиент функции потерь по предсказаниям $F_\omega(\mathbf{s})$ вычисляется как

$$g_i = \frac{\partial L(\mathbf{q}_i), F_\omega(\mathbf{s}_i))}{\partial F_\omega(\mathbf{s}_i)} = F_\omega(\mathbf{s}_i) - \mathbf{q}_i,$$

а гессиан (вторая производная) – как

$$k_i = \frac{\partial^2 L(\mathbf{q}_i), F_\omega(\mathbf{s}_i))}{\partial F_\omega(\mathbf{s}_i)^2} = 1.$$

Каждое новое дерево $h_m(s)$ обучается на остаточных ошибках g_i, минимизируя целевую функцию

$$\text{Obj} = \sum_{i=1}^{N} \left(g_i h_m(\mathbf{s}_i) + \frac{1}{2} k_i h_m(\mathbf{s}_i)^2 \right) + \Theta(h_m),$$

где $\Theta(h_m)$ – регуляризационный член, контролирующий сложность дерева [4]. Итоговый прогноз ансамбля формируется как взвешенная сумма предсказаний всех деревьев:

$$F_\omega(\mathbf{s}) = \sum_{m=1}^{M} \eta h_m(\mathbf{s}),$$

где η – коэффициент скорости обучения, а M – общее число деревьев.

6. Нейронная сеть. Рассмотрим полносвязную нейронную сеть с функцией активации ReLU (Rectified Linear Unit):

$$z^{(l)} = W^{(l)} a^{(l-1)} + b^{(l)},$$

где $W^{(l)}$ – матрица весов, $b^{(l)}$ – вектор смещения, а $a^{(l-1)}$ – выход предыдущего слоя [5]. Затем применяется функция активации ReLU

$$a^{(l)} = \sigma(z^{(l)}) = \max(0, z^{(l)}).$$

На выходном слое используется линейная активация для получения предсказаний углов сочленений $\mathbf{q}$. Нейронная сеть обучается с использованием метода обратного распространения ошибки и оптимизатора Adam.

7. Архитектура робота-манипулятора. Решение прямой задачи. Создание обучающей выборки. Пусть задан робот-манипулятор с семью степенями свободы (см. рис. 1).

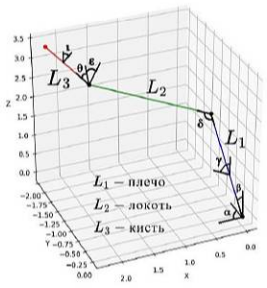


Рис. 1. Схема робота-манипулятора

Манипулятор состоит из трех суставов и трех сочленений ненулевой длины:

- плечевой сферический сустав, углы Эйлера α, β, γ;
- локтевой поворотный сустав, угол δ;
- кистевой сферический сустав, углы $\varepsilon, \theta, \iota$.

Данная схема робота напоминает человеческую руку без пальцев. На углы накладываются ограничения, указанные в таблице.

Таблица. Ограничения на углы

Угол	**Min**	**Max**
α	$-\pi/2$	$\pi/4$
β	0	π
γ	$-\pi/2$	$\pi/2$
δ	0	π
ε	$-\pi/2$	$\pi/2$
θ	$-\pi/2$	$\pi/2$
γ	0	2π

Они обеспечивают работу робота только в полупространстве перед собой. Для данной архитектуры манипулятора можно легко выписать решение прямой задачи кинематики, если принять во внимание тот факт, что углы в сочленениях – углы Эйлера. Воспользуемся

двумя матричными выражениями: поворот вокруг оси z на угол φ:

$$R_z(\varphi) = \begin{pmatrix} \cos(\varphi) & -\sin(\varphi) & 0 \\ \sin(\varphi) & \cos(\varphi) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

и аналогичный поворот вокруг оси x:

$$R_x(\varphi) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\varphi) & -\sin(\varphi) \\ 0 & \sin(\varphi) & \cos(\varphi) \end{pmatrix}.$$

Тогда решение прямой задачи кинематики, если сочленения имеют длины L_1, L_2, L_3 соответственно, для данной схемы имеет вид для координат:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = R_z(\alpha)R_x(\beta)R_z(\gamma) \begin{pmatrix} 0 \\ 0 \\ L_1 \end{pmatrix} + R_x(\delta) \begin{pmatrix} 0 \\ 0 \\ L_2 \end{pmatrix} + \\ + R_z(\varepsilon)R_x(\theta)R_z(\iota) \begin{pmatrix} 0 \\ 0 \\ L_1 \end{pmatrix}.$$

Для нахождения ориентации кисти возьмем орт направляющего вектора последнего сочленения. Для этого рассмотрим последний столбец результирующей матрицы преобразований

$$R = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} = R_z(\alpha)R_x(\beta)R_z(\gamma)R_x(\delta)R_z(\varepsilon)R_x(\theta)R_z(\iota),$$

где в столбцах содержатся координаты ортов системы координат инструментва в базовой системе координат.

Таким образом, получена функция, которая каждому вектору углов сочленений $(\alpha, \beta, \gamma, \delta, \varepsilon, \theta, \iota)^T$ ставит в соответствие единственным образом вектор положения и ориентации рабочего органа $(X, Y, Z, r_{11}, r_{22}, r_{33})^T$, что представляет собой решение прямой задачи кинематики для данной схемы.

Для создания обучающей выборки, обучения, тестирования и тюнинга гиперпараметров был написан алгоритм на Python, с помощью

которого был сгенерирован набор случайных равномерно распределенных углов, и решена прямая задача кинематики. В рамках данной работы введено ограничение: угол собственного вращения плеча в обучающей выборки $\gamma = 0$. Размер обучающей выборки составил 30000 примеров.

8. Обучение и подбор гиперпараметров. Результаты. Все модели были обучены и настроены с использованием GridSearchCV и кросс-валидации на 3 фолда, что позволило обеспечить надежность оценки их обобщающей способности. На рис. 2 приведена визуализация траектории движения рабочего органа манипулятора по прямой линии, полученная методом случайного леса, которая подтверждает корректность работы моделей и их способность решать задачу обратной кинематики.

Для случайного леса оптимальные гиперпараметры, найденные в результате GridSearchCV, оказались следующими:

- количество деревьев: 1000;
- максимальная глубина: 14.

Модель показала ошибки на тестовой выборке MSE: 1,0381, несмотря на его простоту и быстроту вычисления.

Для XGBoost были найдены следующие оптимальные гиперпараметры:

- количество деревьев: 2000;
- максимальная глубина: 12;
- скорость обучения: 0,01.

Модель на тестовых данных достигла ошибки MSE: 1,042, что является худшим из представленных результатом. Стоит отметить, что до подбора гиперпараметров для всех моделей, именно бустинговый алгоритм показал наилучший результат 1,1448.

Для полносвязной нейронной сети были выбраны следующие параметры:

- количество слоев: 2;
- количество нейронов в скрытых слоях: 256;
- скорость обучения: 0,005.

Модель достигла ошибки на тестовых данных (MSE): 1,0114. Нейронная сеть показала наилучшую точность среди всех методов, что

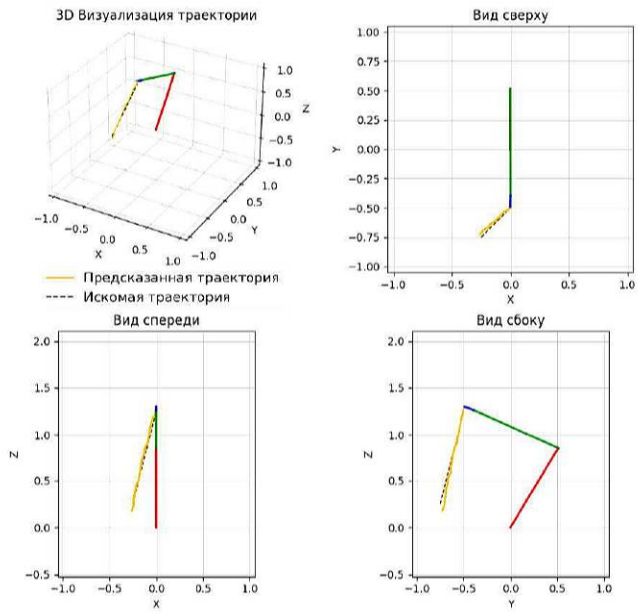


Рис. 2. Визуализация траектории прямой линии

подтверждает ее способности к выявлению сложных нелинейных зависимостей.

9. Заключение. В данной работе рассмотрены три метода машинного обучения для решения задачи обратной кинематики манипулятора с излишним числом степеней свободы: случайный лес, бустинговый алгоритм XGBoost и полносвязная нейронная сеть. Наилучшую точность продемонстрировала нейронная сеть, что подтверждает ее способность аппроксимировать сложные нелинейные зависимости. Метод XGBoost и случайный лес показали ошибку хуже, что делает их менее точными, но более вычислительно эффективными.

Согласно исследованию Эноха и Салема [6], машинное обучение в задачах обратной кинематики может демонстрировать ограниченную точность, особенно в случаях сложных конфигураций манипулятора. Однако даже приближенное решение, полученное с помо-

щью методов машинного обучения, может служить хорошим начальным приближением для оптимизационных методов, таких как метод Ньютона – Рафсона. Эти методы позволяют достичь высокой точности за счет итеративного уточнения решения, что особенно важно в задачах, требующих точного позиционирования рабочего органа.

Таким образом, комбинированный подход, использующий машинное обучение для начального приближения и оптимизационные методы для уточнения, может быть эффективным решением для задач обратной кинематики о положении.

Литература

1. Обада М. Х., Шиманчук Д. В. Использование метода отжига для решения обратной задачи о положении робота-манипулятора // Процессы управления и устойчивость. 2023. Т. 10. № 1. С. 132–139.

2. Chen T., Chen H. Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its applications to dynamic systems // IEEE Transactions on Neural Networks. 1995. Vol. 6. P. 911–917.

3. Lee T., Ullah A., Wang R. Bootstrap aggregating and random forest // Macroeconomic Forecasting in the Era of Big Data. 2020. Vol. 52. P. 389–429.

4. Introduction to boosted trees [Электронный ресурс]: URL:https://xgboost.readthedocs.io/en/stable/tutorials/model.html (дата обращения: 13.12.24).

5. Islam M., Chen G., Jin S. An overview of neural network // American Journal of Neural Networks and Application. 2019. Vol. 5. No 1. P. 7–11.

6. Machine learning and optimization techniques for solving inverse kinematics in a 7-DOF robotic arm [Электронный ресурс]: URL: https://arxiv.org/pdf/2406.13064 (дата обращения: 12.02.25).

7. Ефимова П. А., Алферов Г. В., Иванов Г. Г. Динамическая модель космического манипуляционного робота // Процессы управления и устойчивость. 2015. Т. 2. № 1. С. 173–179.

УДК 629.78

Сулейманов В. Н.

Моделирование формы солнечного паруса под действием давления солнечного света в окрестности точки либрации L1

Рекомендовано к публикации доцентом Шиманчуком Д. В.

1. Введение. В последнее время космические аппараты (КА) с солнечным парусом становятся всё более интересны учёным, особенно движение и управление такими аппаратами в окрестностях точек либрации, где, в силу модели ограниченной задачи трёх тел, гравитационные силы Солнца и Земли уравнивают друг друга с помощью центростремительных сил [1–3]. Таким образом, влияние силы тяги, создаваемой солнечным давлением на парус посредством передачи импульса фотонов, становится определяющей для нахождения траекторий движения. Так как солнечный парус являет собой тонкую плёнку, то под действием давления солнечного света неизбежно будет происходить деформация плёнки, которая ведёт к потерям силы тяги по отношению представления паруса как неизменяемой плоскости. В данной работе рассмотрена оценка деформации солнечных парусов КА (Helianthus Solar Sail, LightSail) под действием давления солнечного света в окрестности коллинеарной точки либрации $L1$ с помощью метода конечных элементов (МКЭ). Влияние гравитационных и центростремительных сил опущено в силу их малости.

2. Нелинейная динамика паруса. Движение деформируемого тела (см. рис. 1), согласно работе [4], определяемого в пространстве радиус-вектором $\mathbf{R}$, описывается вторым законом Ньютона:

$$\iiint\limits_V \mathbf{f}_e \, \mathrm{d}m = \iiint\limits_V \ddot{\mathbf{R}} \, \mathrm{d}m$$

Сулейманов Вадим Наильевич – магистрант, Санкт-Петербургский государственный университет; e-mail: st077208@student.spbu.ru, тел.: +7(912)906-94-13

или

$$\iiint\limits_V (\mathbf{p}+\mathbf{u}) \times \ddot{\mathbf{R}}\, \mathrm{d}m = \iiint\limits_V (\mathbf{p}+\mathbf{u}) \times \mathbf{f}_e\, \mathrm{d}m, \quad (1)$$

где $\mathbf{p}$ и $\mathbf{u}$ – положение в локальной системе координат и деформация массы $\mathrm{d}m$ соответственно. А уравнение обобщённых сил упругости имеет вид

$$\iiint\limits_V \mathbf{\Psi}_\alpha^T \mathbf{f}_e\, \mathrm{d}m = \iiint\limits_V \mathbf{\Psi}_\alpha^T \ddot{\mathbf{R}}\, \mathrm{d}m + \mathbf{f}_\alpha, \quad \alpha = 1, 2, \ldots, \quad (2)$$

где $\mathbf{\Psi}_\alpha$ – функция формы, используемая в МКЭ, $\mathbf{f}_e$ – внешние силы, $\mathbf{f}_\alpha$ – обобщённые силы упругой деформации, связанные с функцией формы $\mathbf{\Psi}_\alpha$.

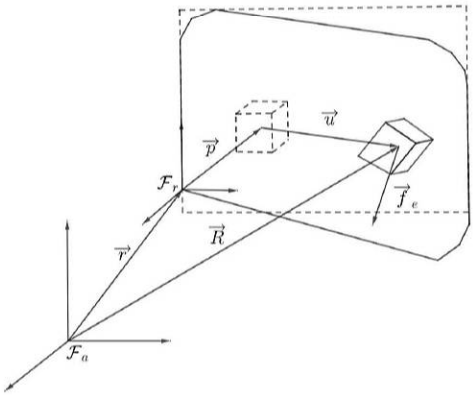


Рис. 1. Нестеснённое деформируемое тело объёма V

Не предполагая малых деформаций, уравнения (1),(2) можно переписать в виде

$$\begin{bmatrix} M_{rr}(q_e) & M_{re}(q_e) \\ M_{re}^T(q_e) & M_{ee} \end{bmatrix} \begin{bmatrix} \ddot{\xi} \\ \ddot{q}_e \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & K_{ee}(q_e) \end{bmatrix} \begin{bmatrix} \gamma \\ q_e \end{bmatrix} =$$

$$= \begin{bmatrix} f_{T,r}(q) \\ f_{T,e} \end{bmatrix} + \begin{bmatrix} f_{I,r}(q) \\ f_{I,e}(q) \end{bmatrix},$$

где $\xi = [v^T \quad \omega^T]$, $\dot{q} = [\dot{\xi}^T \quad \dot{q}_e^T]$, $q = [\gamma^T \quad q_e^T]^T$ – фазовые переменные,

v и ω – линейная и угловая скорости локальной системы координат $\mathcal{F}_r$ относительно глобальной системы координат $\mathcal{F}_a$ соответственно; вектор γ – фиктивная переменная, удовлетворяющая $\dot{\gamma} = \xi$; матрица M – матрица масс, составления из матриц масс твёрдого тела M_{rr}, M_{re} и M_{ee}, которые являются условиями связи поведения твёрдого и упругого тела; K_{ee} – матрица жёсткости упругого тела; векторы f_T и f_I – внешние и инерционные силы твёрдого r и упругого e тела соответственно.

Согласно теории пластин Кирхгофа – Лява [5,6], деформации, зависящие от смещения в связанной с телом декартовой прямоугольной системе координат, аппроксимируются как

$$u = u_1\overrightarrow{e_1} + u_2\overrightarrow{e_2} + u_3\overrightarrow{e_3},$$

$$\varepsilon_{\alpha\beta} = \frac{1}{2}(u_{\alpha,\beta} + u_{\beta,\alpha} + \psi_\alpha\psi_\beta) - x_3\psi_{\alpha\beta},$$

$$\varepsilon_{\alpha 3} = -\psi_\alpha + \psi_\alpha = 0, \qquad \varepsilon_{zz} = 0,$$

$$u_{\alpha,\beta} = \frac{\partial u_\alpha}{\partial x_\beta}, \quad \alpha,\beta = 1,2,$$

где ψ_α – углы поворота нормали к срединной поверхности пластины при деформации.

Для изотропной и однородной пластины зависимости напряжения от деформации имеют вид

$$\begin{bmatrix}\sigma_{11}\\ \sigma_{22}\\ \sigma_{12}\end{bmatrix} = \frac{E}{1-\nu^2}\begin{bmatrix}1 & \nu & 0\\ \nu & 1 & 0\\ 0 & 0 & 1-\nu\end{bmatrix}\begin{bmatrix}\varepsilon_{11}\\ \varepsilon_{22}\\ \varepsilon_{12}\end{bmatrix}, \qquad (3)$$

где E – модуль Юнга, ν – коэффициент Пуассона. Моменты, соответствующие данным напряжениям:

$$\begin{bmatrix}M_{11}\\ M_{22}\\ M_{12}\end{bmatrix} = -\frac{2h^3E}{3(1-\nu^2)}\begin{bmatrix}1 & \nu & 0\\ \nu & 1 & 0\\ 0 & 0 & 1-\nu\end{bmatrix}\begin{bmatrix}\psi_{11}\\ \psi_{22}\\ \psi_{12}\end{bmatrix}, \qquad (4)$$

где $2h$ – толщина пластины.

При чистом изгибе, т. е. отсутствии поперечных сил, при дифференцировании уравнений (3), (4) и подстановки результатов в условия равновесия пластины, полученные из принципа виртуальной работы, основное уравнение пластины принимает бигармонический вид

$$\nabla^2\nabla^2\psi = 0.$$

Условия равновесия тонкой пластины при квазистатическом прогибе, в силу принципа виртуальной работы, при действии поперечной нагрузки имеют вид

$$\frac{\partial N_{11}}{\partial x_1} + \frac{\partial N_{21}}{\partial x_2} = 0, \quad \frac{\partial N_{12}}{\partial x_1} + \frac{\partial N_{22}}{\partial x_2} = 0,$$

$$\frac{\partial^2 M_{11}}{\partial^2 x_1} + 2\frac{\partial^2 M_{12}}{\partial x_1 \partial x_2} + \frac{\partial^2 M_{22}}{\partial^2 x_2} = q,$$

$$N_{\alpha\beta} = \int\limits_{-h}^{h} \sigma_{\alpha\beta}\,\mathrm{d}x_3, \quad M_{\alpha\beta} = \int\limits_{-h}^{h} x_3\sigma_{\alpha\beta}\,\mathrm{d}x_3,$$

где $\sigma_{\alpha\beta}$ – механические напряжения, q – поперечная нагрузка на пластину.

В таком случае уравнение пластины принимает вид

$$\nabla^2\nabla^2\psi = \frac{q}{D}, \quad D = \frac{2h^3E}{3(1-\nu^2)}.$$

3. Постановка задачи. Рассмотрим солнечный парус как тонкую пластину, ориентированную прямо на солнце в окрестности точки либрации $L1$, находящейся на расстоянии одной астрономической единицы от Солнца. Давление солнечного света на данном расстоянии $P = 4{,}56 \times 10^{-6}$ Па. Влияние гравитационных и центростремительных сил опущено. Таким образом, задача сводится к квазистатическому прогибу тонкой пластины под действием поперечной нагрузки.

4. Характеристики солнечных парусов. Конструкции солнечных парусов Helianthus Solar Sail, LightSail представим как квадратные пластины. Начало локальной системы координат поместим в центр квадрата в место крепления КА к парусу (см. рис. 2).

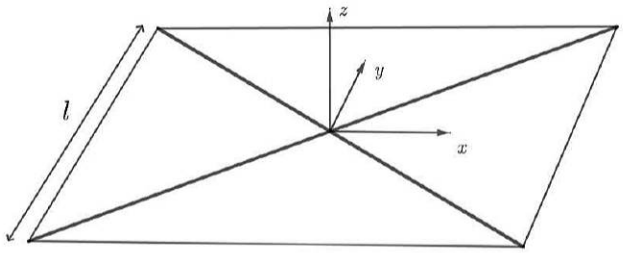


Рис. 2. Конструкция солнечного паруса

Таким образом, граничные условия – штанги солнечного паруса, к которым крепится полотно.

Граничные условия представимы в виде

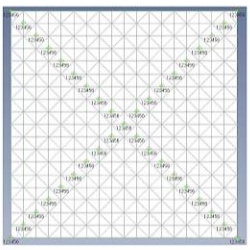
$$\Omega = \{x = y \ \cap \ z = 0; \ x = -y \ \cap \ z = 0\}, \ \psi|_{\partial\Omega} = 0, \ \frac{\partial\psi}{\partial x_\alpha}\Big|_{\partial\Omega} = 0.$$

Характеристики солнечных парусов взяты из [7–9] и представлены в таблице.

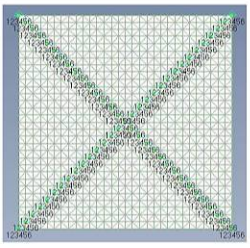
Таблица. Характеристики солнечных парусов

№	**Helianthus Solar Sail**	**LightSail**
сторона паруса l, м	35,58	5,6
толщина h, м	$2{,}5 \times 10^{-4}$	1×10^{-4}
E, ГПа	2,48	2,48
ν	0,34	0,34

5. Численное моделирование. Моделирование проводится методом конечных элементов независимо на языке Python с помощью библиотеки scikit-fem [10] и в САПР Siemens Femap with NX Nastran 2401. Солнечные паруса разбиваются простыми треугольными элементами. В Python модели Helianthus Solar Sail и LightSail разбиваются на 1024 элемента; в Femap – на 648 и 1800 элементов соответственно (см. рис. 3). В результате вычислений получено, что максимальный прогиб для Helianthus Solar Sail в Python составил 1,137 метра; в Femap – 1,166 метра (см. рис. 4). Для паруса LightSail в Python – 0,0152 метра; в Femap – 0,0147 метра (см. рис. 5).

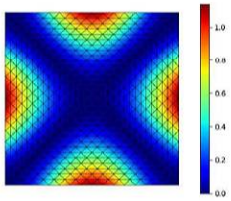


Helianthus Solar Sail

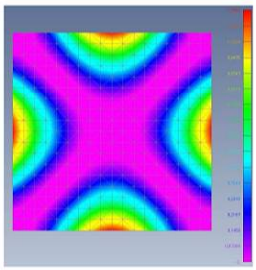


LightSail

Рис. 3. Сетка из треугольных элементов с граничными условиями в Femap

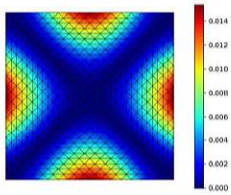


scikit-fem Python

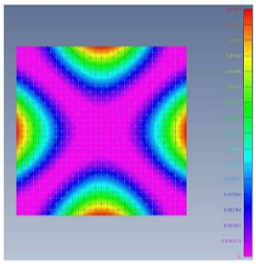


Femap

Рис. 4. Значение прогиба солнечного паруса Helianthus Solar Sail



scikit-fem Python



Femap

Рис. 5. Значение прогиба солнечного паруса LightSail

6. Выводы. В работе рассмотрен квазистатичный прогиб солнечного паруса в окрестности коллинеарной точки либрации $L1$. Анализируя полученные результаты, можно сделать вывод, что чем больше полотно паруса, тем больше при рассмотренных граничных условиях величина прогиба, что может приводить к большим потерям силы тяги, создаваемой солнечным давлением, по сравнению с представлением паруса не деформируемой пластиной. Если учитывать силы Кориолиса, появляющиеся при управляемом движении, совместно с моментами, создаваемыми изменением ориентации паруса, то задача значительно усложняется рассмотрением динамических уравнений Кирхгофа – Лява для пластин, где возможны колебания паруса, которые могут привести к конструкционным изменениям, что упомянуто в [11].

Литература

1. Шиманчук Д. В., Шмыров А. С., Шмыров В. А. Управляемое движение солнечного паруса в окрестности коллинеарной точки либрации // Письма в астрономический журнал. 2020. Т. 46 № 3. С. 193–200.

2. Шиманчук Д. В., Шмыров В. А. Построение траектории возвращения в окрестность коллинеарной точки либрации системы Солнце-Земля // Вестник Санкт – Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2013. № 2. С. 76–85.

3. Шмыров А. С., Шмыров В. А., Шиманчук Д. В. Управление орбитальным движением космического аппарата вблизи точки либрации с помощью изменения отражательной способности солнечного паруса // Вестник Санкт – Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2024. Т. 20 № 2. С. 298—306.

4. Choi M., Damaren C. J. Structural dynamics and attitude control of a solar sail using tip vanes // Journal of Spacecraft and Rockets. 2015. Vol. 52. No 6. P. 1665–1679.

5. Мяченков В. И., Мальцев В. П., Майборода В. П. и др. Расчеты машиностроительных конструкций методом конечных элементов М.: Машиностроение, 1989. 520 с.

6. Love A. E. H. On the small free vibrations and deformations of elastic shells // Philosophical Transactions of the Royal Society. 1888. Vol. serie A. No 17. P. 491–549.

7. Boni L., Basetto M., Niccolai L. et al. Structural response of Helianthus solar sail during attitude maneuvers // Aerospace and Technology. 2023. No 133. P. 108152.

8. LightSail, a Planetary Society solar sail spacecraft // The Planetary Society [Электронный ресурс]: URL:`https://www.planetary.org/sci-tech/lightsail` (дата обращения: 01.03.2025).

9. Kapton Summary of Properties [Электронный ресурс]: URL:`https://www.dupont.com/content/dam/electronics/amer/us/en/electronics/public/documents/en/EI-10142_Kapton-Summary-of-Properties.pdf` (дата обращения: 03.03.2025).

10. Documentation of scikit-fem [Электронный ресурс]: URL:`https://scikit-fem.readthedocs.io/en/stable/` (дата обращения: 20.02.2025).

11. Чекашов А. С., Алипова Б. Н., Старинова О. Л. Моделирование формы поверхности солнечного паруса // Сб. тр. «Министерство образования и науки Российской Федерации». Самара: Изд-во Самарского университета, 2017. С. 70–71.

УДК 539.421

Якимова И. О., Шувалов Г. М.

Моделирование распространения трещины под действием напряжений и коррозии

1. Введение. Трещины, практически всегда присутствующие в реальных конструкционных материалах, могут существенно изменять их упругие эффективные свойства [1, 2]. Выступая концентраторами напряжения, они создают зоны локального перенапряжения и могут приводить к внезапному разрушению конструкций даже при нагрузках ниже предельно допустимых. Моделирование процесса роста трещин позволяет определить как их критические размеры, при достижении которых происходит лавинообразный рост трещин и разрушение, так и время, за которое эти размеры будут достигнуты. Это дает возможность перейти от реактивного обслуживания элементов конструкций, т. е. по факту их поломки, к превентивному и плановому, снижая риски и затраты.

Рост трещин в конструкциях может быть обусловлен различными факторами: циклическими нагрузками, коррозионными процессами, усталостными воздействиями или комбинацией всего перечисленного. В данной работе исследуется коррозионный рост радиальной трещины, исходящей из кругового отверстия в тонкой стальной пластине, под действием одноосного растяжения.

2. Постановка задачи. Рассматривается изотропная тонкая прямоугольная пластина размерами $W \times H$, ослабленная центральным отверстием радиуса r и находящаяся в условиях плоского напряженного состояния. Из отверстия исходит радиальная трещина начальной длины $a = a(0)$, направленная вдоль продольной оси пластины (см. рис. 1). К границам пластины, параллельным направлению трещины, приложены постоянные растягивающие усилия p:

$$\sigma_{yy}(x, y)|_{y=\pm H/2} = p, \quad \sigma_{xx}(x, y)|_{y=\pm H/2} = \sigma_{xy}(x, y)|_{y=\pm H/2} = 0,$$

Якимова Ирина Олеговна – студент, Санкт-Петербургский государственный университет; e-mail: st117045@student.spbu.ru, тел.: +7(913)562-63-66

Шувалов Глеб Михайлович – доцент, Санкт-Петербургский государственный университет; e-mail: g.shuvalov@spbu.ru, тел.: +7(921)387-58-63

Работа выполнена при финансовой поддержке РНФ, проект № 25-29-00565

где σ_{ij} – компоненты тензора напряжений в декартовой прямоугольной системе координат Oxy, центр которой совпадает с центром отверстия, а ось Ox направлена вдоль трещины.

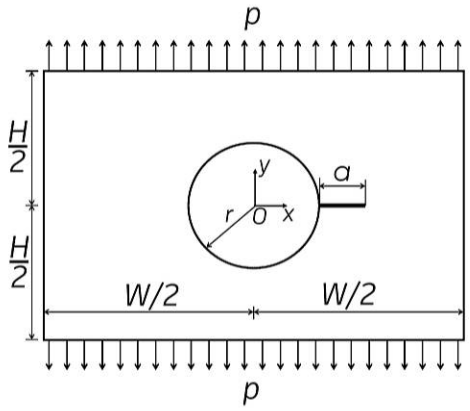


Рис. 1. Пластина, ослабленная вырезом и трещиной

Требуется определить $a(t)$ – зависимость длины трещины от времени в условиях коррозии под напряжением в предположении, что скорость коррозии пропорциональна локальным напряжениям у вершины трещины.

В рамках данной работы для описания роста трещины используется уравнение, полученное из рассмотрения процессов коррозионного растворения материала вблизи вершины трещины [3]:

$$\frac{da}{dt} = 1{,}1 \cdot 10^{-7} \times$$
$$\times \left[2{,}5 \cdot 10^{10} \cdot \exp\left(\frac{3 \cdot 10^{-19} - 1{,}5 \cdot 10^{-20} (K_I - K_{ISCC})^{1/3}}{7{,}74 \cdot 10^{-21}} \right) \right]^{0{,}433}. \quad (1)$$

Здесь коэффициент интенсивности напряжений K_I и его пороговое значение K_{ISCC} имеют размерность МПа $\cdot \sqrt{\text{м}}$, скорость роста трещины – м/с. При этом считается, что при $K_I \leqslant K_{ISCC}$ трещина не развивается. При достижении величиной K_I критического значения K_{IC} (вязкость разрушения) происходит лавинообразный рост трещины и мгновенное разрушение.

Отметим, что в данной работе рассматривается только режим

устойчивого роста трещины ($K_{ISCC} < K_I < K_{IC}$). Пороговое значение коэффициента интенсивности напряжений, входящее в уравнение (1), принято равным $K_{ISCC} = 2$ МПа$\cdot\sqrt{\text{м}}$ [3], величина K_{IC} для аустенитных сталей лежит в диапазоне 80–200 МПа$\cdot\sqrt{\text{м}}$.

3. Методы решения. Ввиду отсутствия аналитического решения соответствующей краевой задачи теории упругости, перемещения границы трещины в рамках данного исследования определяются методом конечных элементов. Для интегрирования уравнения роста трещины (1) необходимо в каждый момент времени знать коэффициент интенсивности напряжений вблизи вершины трещины K_I, который, в свою очередь, зависит от длины трещины a и номинальных напряжений. В данной работе для решения уравнения (1) используется уточненный итерационный метод Эйлера с пошаговым контролем точности. Коэффициент интенсивности напряжений на каждом шаге вычисляется методом экстраполяции перемещений [4, 5]:

$$K_I = \lim_{r\to 0} \frac{\sqrt{2\pi}G}{1+\kappa} \frac{|\Delta v(r)|}{\sqrt{r}},$$

где $|\Delta v(r)|/\sqrt{r}$ считается линейной функцией расстояния r от вершины трещины до точек на ее границе. Причем для рассматриваемой симметричной задачи $|\Delta v(r)| = 2v(r)$, $v(r)$ – перемещение в направлении, перпендикулярном плоскости трещины, $\kappa = (3-\nu)/(1+\nu)$ при плоском напряженном состоянии, G и ν – модуль сдвига и коэффициент Пуассона материала пластины.

4. Численные результаты. В качестве примера рассмотрена тонкая пластина размерами $W = 0{,}6$ м и $H = 0{,}8$ м. Упругие параметры материала пластины соответствуют стали: модуль Юнга $E = 200$ ГПа, коэффициент Пуассона $\nu = 0{,}3$.

При определении напряженно-деформированного состояния методом конечных элементов рассматривалась половина пластины, разделенная плоскостью трещины, вдоль которой задавались условия симметрии (нулевые перемещения в направлении, перпендикулярном к границе). На противоположной стороне пластины прикладывались растягивающие усилия. Пластина моделировалась плоскими треугольными шестиузловыми элементами. Для повышения точности решения конечно-элементная сетка сгущалась вблизи отверстия и фронта трещины. В окрестности вершины трещины использовались сингулярные элементы, учитывающие асимптотику напря-

жений. На каждой итерации метода Эйлера по значениям перемещений в двух ближайших к вершине трещины узлах вычислялся коэффициент интенсивности напряжений K_I и с использованием уравнения (1), определялась новая длина трещины.

На рис. 2 представлена зависимость длины трещины от времени для различных радиусов отверстия $r = \{25; 50\}$ мм и растягивающих усилий $p = \{50; 100\}$ МПа.

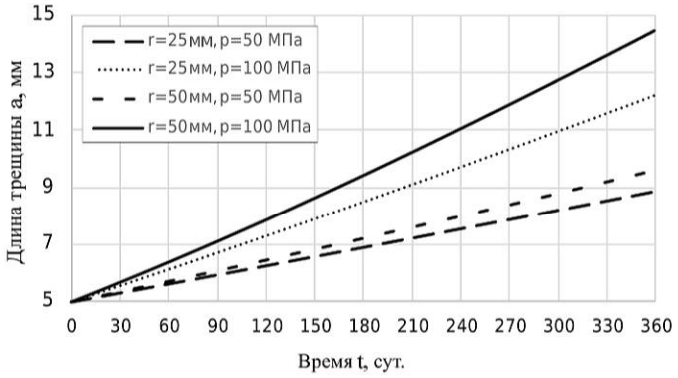


Рис. 2. Зависимость длины трещины a от времени t

В дальнейшем будет исследовано влияние формы отверстия и размеров пластины на скорость роста трещины. Кроме того, с использованием разработанной ранее методики [6], планируется учесть поверхностную упругость с целью уточнения решения для случая наноразмерных отверстий и трещин. Отметим, что предложенный подход также может быть обобщен для исследования морфологической устойчивости пор при диффузионных процессах [7] и коррозионного растворения тонких оболочек под напряжением [8].

5. Заключение. В данной работе разработан алгоритм решения задачи о прямолинейном росте радиальной трещины, исходящей из отверстия, под действием напряжений и коррозии. Скорость роста трещины считалась зависящей от коэффициента интенсивности напряжений, который в рамках линейно-упругого анализа в каждый момент времени определялся экстраполяцией перемещений с использованием метода конечных элементов. Рассмотрен пример и получены зависимости длины трещины от времени для различных радиусов окружностей и растягивающих усилий.

Литература

1. Абакаров А. М., Пронина Ю. Г. Об эффективных упругих свойствах материала со взаимно перпендикулярными системами параллельных трещин // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2022. Т. 18. № 1. С. 111–119.

2. Вяльцева В. В., Пронина Ю. Г. Об оценке трехмерной плотности параллельных трещин по их следам на поверхности // Процессы управления и устойчивость. 2024. Т. 11. No 1. С. 89–92.

3. Fujii T., Tohgo K., Kenmochi A., Shimamura Y. Experimental and numerical investigation of stress corrosion cracking of sensitized type 304 stainless steel under high-temperature and high-purity water // Corrosion Science. 2015. Vol. 97. P. 139–97.

4. Качанов Л. М. Основы механики разрушения. М.: Наука, 1974. 312 с.

5. Zhu N., Oterkus E. Calculation of stress intensity factor using displacement extrapolation method in peridynamic framework // Journal of Mechanics. 2020. Vol. 36. No 2. P. 235–243.

6. Shuvalov G. M., Vakaeva A. B., Shamsutdinov D. A., Kostyrko S. A. The effect of nonlinear terms in boundary perturbation method on stress concentration near the nanopatterned bimaterial interface // Vestnik of Saint Petersburg University. Applied Mathematics. Computer Science. Control Processes. 2020. Vol. 97. P. 165–176.

7. Вакаева А. Б., Шувалов Г. М., Костырко С. А. Морфологическая неустойчивость нанопоры в твердом теле при поверхностной диффузии // Процессы управления и устойчивость. 2021. Т. 8. No 1. С. 108–112.

8. Evstafeva I., Pronina Y. On the mechanochemical dissolution of shells and its temperature dependence: discussion of different models // International Journal of Engineering Science. 2023. Vol. 190. Art. no 103889.

UDC 519.8

Li Y.

Design and implementation of Multi-UAV cooperative reconnaissance and strike mission planning system

Recommended for publication by associate professor Eremin A. S.

1. Introduction. With the rapid advancement and extensive application of unmanned aerial vehicle (UAV) technology, particularly in domains such as military operations, emergency rescue [1], and environmental monitoring, the research and application of multi-UAV systems have increasingly become a focal point of interest. In comparison to single UAVs, multi-UAV systems offer significant advantages, including high mission execution efficiency, broad coverage areas, and robust mission adaptability. Through the collaborative efforts of multiple UAVs, more complex and challenging tasks can be accomplished, such as large-scale surface reconnaissance, multi-target tracking and engagement, and rapid response to emergencies [2]. However, as task complexity increases, how to conduct scientific and reasonable task planning within infrastructure constraints [3] and achieve efficient collaboration among UAVs has emerged as a critical technical challenge that urgently requires resolution.

In the research on multi-unmanned aerial vehicle (UAV) cooperative reconnaissance and strike technology, numerous scholars both domestically and internationally have conducted in-depth analyses of its key challenges from various perspectives, thereby advancing the theoretical development and engineering applications in this domain. Regarding task allocation and path planning, Matarić [4] proposed as early as in 2002 a multi-robot collaborative approach based on behavioral models, emphasizing the coupling mechanism between local perception and collective behavior within autonomous systems. In recent years, the team led by How [5] at MIT has developed a multi-unmanned aerial vehicle (UAV) task allocation framework utilizing game-theoretic methods and mixed integer linear programming (MILP), achieving efficient division of

Li Yupeng – undergraduate student, Yan'an University,
email: 1935954699@qq.com

labor and path optimization for heterogeneous unmanned systems under complex operational scenarios.

2. System design and implementation. According to the multi-UAV collaborative area search and strike mission planning for this project, the designed UAV payloads primarily consist of reconnaissance payloads and strike payloads. A detailed analysis of the signal characteristic indicators for both types of payloads is presented below.

The reconnaissance payload of the UAV consists of pod-mounted electronic or electro-optical equipment, which achieves scanning and reconnaissance during flight by adjusting the device's attitude.

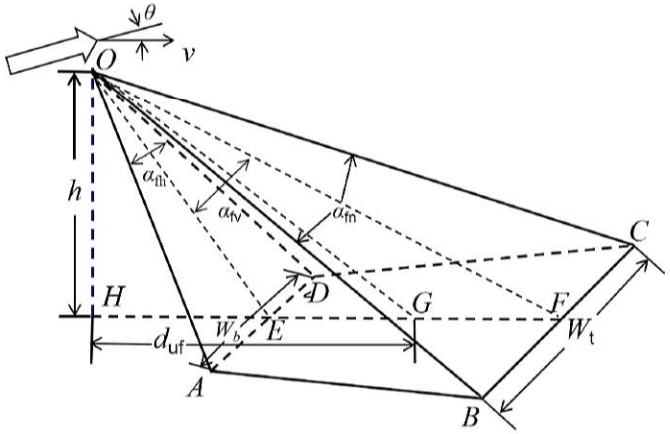


Figure 1. Schematic diagram of UAV reconnaissance payload attitude and scanning rang

As illustrated in Figure 1 showing the UAV reconnaissance payload attitude and scanning range schematic, the UAV operates at altitude h with velocity v. The reconnaissance payload system involves critical angular parameters: aircraft pitch angle θ, payload mounting angle α_m (shown in Fig. 2), load scanning rear Angle α_{fh} and load scanning rake Angle α_{fn}.

Figure 2's side and top views of the UAV reconnaissance payload's attitude and scanning range provide the velocity-direction length (l_{f}) of the search area and scanning widths (W_b, W_t) through geometric analysis.

Based on geometric principles, the following calculations are per-

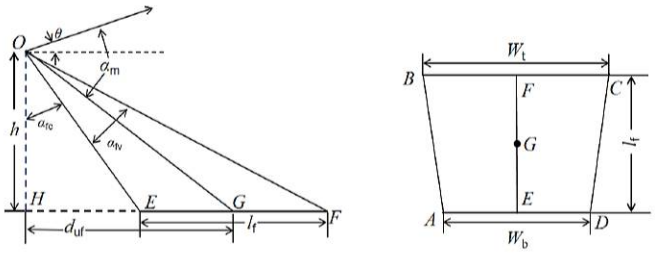


Figure 2. Side view of UAV reconnaissance payload attitude and top view of scanning range

formed:

$$d_{\mathrm{uf}} = h/\tan(\alpha_m - \theta), \quad HF = h/\tan(\alpha_{\mathrm{fc}} + \alpha_{\mathrm{fv}}), \quad l_{\mathrm{f}} = 2(HF - d_{\mathrm{uf}}),$$

$$OE = h/\cos(\alpha_{\mathrm{fc}}), \quad W_b = 2OE\tan(\alpha_{\mathrm{fh}}/2), \quad W_t = 2OE\tan(\alpha_{\mathrm{fn}}/2).$$

Based on the above formula, l_{f}, W_b, and W_t can be obtained, and then the trapezoidal scanning area of the load can be obtained.

3. System design and implementation.

3.1. UAV payload analysis. This project aims to design and implement a multi-UAV cooperative reconnaissance and strike mission planning system based on PyQt. This system is oriented towards the demands of collaborative search, tracking and strike by multiple unmanned aerial vehicles in modern combat missions, and provides integrated functions such as mission modeling, path planning and simulation demonstration.

3.2. Algorithm demonstration. This study presents a UAV path planning algorithm implemented in Python using Matplotlib. The system generates a Voronoi diagram from randomly distributed non-obstacle points and predefined obstacles (buildings/threat sources) to map feasible flight zones. Using graph theory principles, we construct an undirected graph model [6] on the Voronoi edges, incorporating start and end points as non-obstacle nodes. Dijkstra's algorithm is then applied to compute the optimal path. The visualization scheme employs: small blue dots for obstacles, large blue dots for ammunition locations, green/red squares for start/end points respectively, and a red line to indicate the computed path, effectively demonstrating the complete path planning methodology and results.

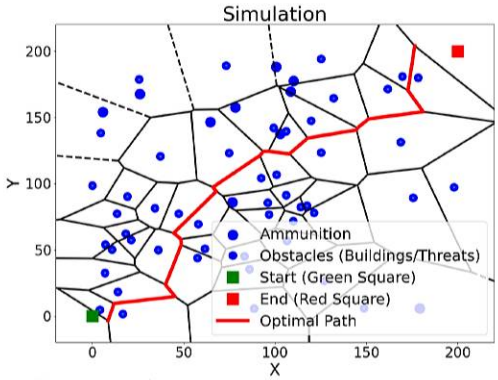


Figure 3. Algorithm demonstration module

3.3. Multi-UAV simulation. This program realizes the path planning and animation simulation of multi-UAV cooperative combat, and is suitable for simulating autonomous navigation and obstacle avoidance tasks in complex environments. The program defines the starting point and target point of the unmanned aerial vehicle (UAV), randomly generates obstacles, and generates a two-dimensional Voronoi diagram as the basis for path search. The Dijkstra algorithm is used to plan the shortest path to avoid obstacles [7]. The UAV moves forward along the path at a speed of 10 meters per frame to generate the trajectory. The animation uses Matplotlib to dynamically display the process of unmanned aerial vehicles crossing the Voronoi area, simulate formation flight, and clearly show obstacles, path planning, and the starting and ending points.

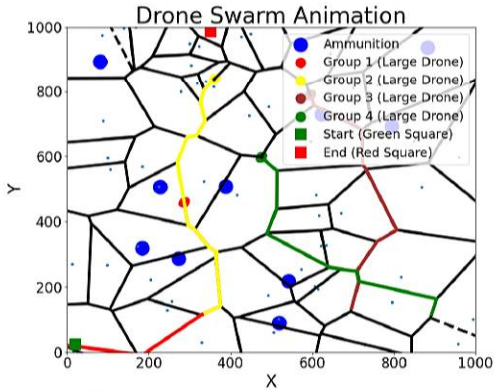


Figure 4. Multi-UAV simulation

4. Conclusion. This system has successfully completed the implementation of the collaborative path planning algorithm guided by Voronoi diagrams. By introducing the Dijkstra algorithm search and graph optimization strategy, it independently completes the optimal path selection from the starting point to the strike target. The system not only supports the modeling of complex terrain environments, but also realizes the collaborative obstacle avoidance of paths among multiple unmanned aerial vehicles, ensuring the safety and efficiency during the mission execution process.

References

1. Jin L. B. Task allocation for multiple UAVs in complex rescue operations // China Science and Technology Information. 2025. No 9. P. 96–98. (In Chinese)

2. Ding X., Jiang H. Y., Guo Y. W., et al. UAV recognition and tracking system for air-to-air scenarios // Industrial Control Computer. 2024. Vol. 37. No 12. P. 6–8. (In Chinese)

3. Outay F., Mengash H. A., Adnan M. Applications of unmanned aerial vehicle (UAV) in road safety, traffic and highway infrastructure management: Recent advances and challenges // Transportation Research Part A: Policy and Practice. 2020. Vol. 141. P. 116–129.

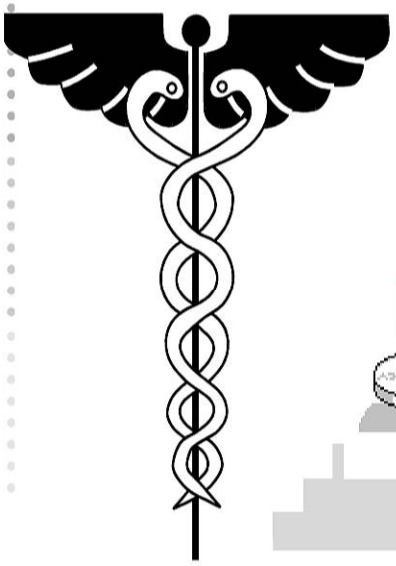
4. Matarić M. J. Designing and understanding adaptive group behavior // Adaptive Behavior. 1995. Vol. 4. No 1. P. 51–80.

5. Alighanbari M., Kuwata Y., How J.P. Coordination and control of multiple UAVs with timing constraints and loitering // Proc. 2003 American Control Conference, Denver, CO, USA. 2003. Vol. 6. P. 5311–5316.

6. Liu Z. M., Zhang C. Research on Hamilton circuit solution for complete undirected graphs based on improved ant colony algorithm // Information and Computer (Theory Edition). 2023. Vol. 35. No 15. P. 123–125. (In Chinese)

7. De Nagy Koves Hrabar S. E. Vision-based three-dimensional navigation for an autonomous helicopter. PhD thesis. USA: University of Southern California, 2006. 193 p.

3. Математические модели медико-биологических систем





УДК 519.246.8

Аникушин В. А., Редько А. Ю.

Прогнозирование динамики эпидемии COVID-19 на основе процентных приростов в странах Европы и Азии

Рекомендовано к публикации профессором Захаровым В. В.

1. Введение. Пандемия COVID-19, вызванная новым вирусом SARS-CoV-2, стала беспрецедентным испытанием для здравоохранения, экономики и социальной сферы в большинстве стран мира. Начиная с конца 2019 года, вирус стремительно распространился по всему миру, были зарегистрированы миллионы случаев заражения и значительное число смертей [1]. Научные исследования, направленные на моделирование распространения инфекционных заболеваний, играют ключевую роль в анализе динамики эпидемий и в разработке эффективных стратегий их контроля. Данная работа посвящена анализу возможностей применения линейной аппроксимации и прогнозных значений процентного прироста для оценки динамики заболеваемости в странах Европы: Италии, Франции и Великобритании, а также в странах Азии: Китае, Японии и Южной Корее.

2. Постановка задачи. В работе исследуются методы прогнозирования динамики общего количества заболевших (C), общего количества выздоровевших и умерших (R), а также числа активных случаев (I) с использованием процентных приростов [2]. Основное внимание уделяется разработке моделей на основе линейной аппроксимации и оценке их эффективности в контексте краткосрочного прогнозирования эпидемиологической ситуации. В данном исследовании рассматривается прогнозирование на две и четыре недели вперёд с использованием прогнозных значений процентных приростов общего количества заболевших (C) и общего количества пациентов,

Аникушин Василий Александрович – магистрант, Санкт-Петербургский государственный университет; e-mail: st087566@student.spbu.ru, тел.: +7(952)373-07-21

Редько Анна Юрьевна – магистрант, Санкт-Петербургский государственный университет; e-mail: st090114@student.spbu.ru, тел.: +7(989)801-17-35

для которых болезнь завершилась (R), а также оценка точности модели путём анализа ошибок прогноза числа активных случаев болезни (I), общего количества заболевших (C) и вычисления значений MAPE (средней абсолютной процентной ошибки) на различных горизонтах прогнозирования. Цель исследования состоит в разработке и оценке эффективности предложенного подхода для краткосрочного прогнозирования эпидемической динамики. В работе проанализированы как точность прогнозов, так и ошибки модели, что позволяет обосновать её применимость для практических задач управления эпидемией.

2.1. Интегральная модель притока и оттока. В соответствии с информацией об общем количестве подтверждённых случаев (C), общем количестве выздоровевших и умерших (R) и числа активных случаев (I) в рассматриваемых странах с момента начала эпидемии до дня t включительно, имеем временной ряд значений общего количества заболевших – интегральный ряд новых ежедневных случаев заболевания $C(t)$ и временной ряд общего количества выздоровевших и умерших – интегральный ряд ежедневных значений суммы выздоровевших и умерших $R(t)$. Обозначая через $\Delta(t)$ количество новых случаев, зарегистрированных в день t, а через $\Delta_1(t)$ – количество выздоровевших и умерших в день t, имеем

$$C(t) = C(t-1) + \Delta(t),$$

$$R(t) = R(t-1) + \Delta_1(t), \quad t = 1, 2 \ldots$$

Процентный прирост интегрального объёма притока (количества выявленных случаев инфицирования) в день t вычисляется по формуле

$$r_{\text{inf}}(t) = \frac{\Delta(t)}{C(t-1)} \cdot 100\%, \tag{1}$$

а процентный прирост интегрального объёма оттока (количества выздоровевших и умерших) в день t

$$r_{\text{of}}(t) = \frac{\Delta_1(t)}{R(t-1)} \cdot 100\%. \tag{2}$$

Поскольку, что $\Delta(t)$ и $\Delta_1(t)$ являются случайными величинами, принимающими неотрицательные значения, процентные приросты, вычисляемые на их основе, также являются неотрицательными случайными величинами.

Воспользуемся формулами (1), (2) и запишем систему дискретных уравнений:

$$I(t) = C(t) - R(t), \tag{3}$$

$$C(t) = \left(1 + \frac{r_{\text{inf}}(t)}{100}\right) C(t-1), \tag{4}$$

$$R(t) = \left(1 + \frac{r_{\text{of}}(t)}{100}\right) R(t-1). \tag{5}$$

Здесь переменная $C(t)$ характеризует интегральный приток (общее количество) новых случаев заболевания в множество активных случаев болезни, а переменная $R(t)$ – интегральный отток суммарного количества выздоровевших и умерших до момента времени t из множества активных случаев болезни. Модель динамики заболевания (3)–(5) принято называть интегральной моделью притока и оттока болеющих людей [3, 4].

2.2. Ретроспективный анализ. Для анализа использовались статистические данные о количестве заболевших, выздоровевших и умерших, предоставленные Центром CSSE Университета Джонса Хопкинса [5]. Для ретроспективного анализа охватывается временной интервал с 22 января 2020 года по 31 декабря 2020 года. Этот период был разбит на двухнедельные и четырёхнедельные промежутки.

Для каждого промежутка, кроме последнего, была выполнена аппроксимация процентных приростов $r_{\text{inf}}(t)$ и $r_{\text{of}}(t)$ с использованием степенной и экспоненциальной функции. Выбор конкретной функции аппроксимации определялся на основе характеристики временного ряда и его поведения на данном интервале. Экспоненциальные функции применялись в случаях, когда временной ряд демонстрировал экспоненциальный рост, в то время как степенные функции использовались для описания более сложных нелинейных зависимостей. При этом был взят за основу метод аппроксимации процентных приростов на временном промежутке, предшествующем дате построения прогноза, длина которого совпадает с горизонтом прогнозирования.

На основе построенной аппроксимирующей функции выполнялась экстраполяция процентного прироста на соответствующий горизонт прогнозирования: для двухнедельных интервалов – на две недели, для четырёхнедельных – на четыре. Для построения прогнозных значений процентных приростов на горизонте прогнозирования использовались построенные экстраполирующие функции. На основе этих прогнозных значений были рассчитаны прогнозные значения временных рядов общего числа заболевших, а также общего числа выздоровевших и умерших. С использованием данных этих временных рядов и преобразованного уравнения (3), было рассчитано прогнозное количество активно болеющих людей за выбранный период.

Для оценки точности прогноза для каждого момента времени рассчитывалась относительная погрешность, сравнивающая прогнозные и фактические значения общего числа заболевших и активных случаев:

$$\delta_t = \frac{\widehat{y}_t - y_t}{y_t} \cdot 100\%,$$

где y_t – это фактическое значение в момент времени t, а $\widehat{y}_t$ – спрогнозированное значение в момент времени t.

Для каждого промежутка считается средняя абсолютная процентная ошибка

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{\widehat{y}_t - y_t}{y_t}\right| \cdot 100\%.$$

3. Результаты работы. Рассмотрим результаты проведённого ретроспективного анализа. В таблицах 1, 2 представлены ошибки MAPE прогнозирования общего количества заболевших при горизонте прогнозирования две и четыре недели соответственно. Можно заметить, что первые промежутки пустые, а прогнозирование начиналось только с третьего промежутка. Это связано с качеством данных на первых промежутках. Также видно, что изначально ошибки MAPE были достаточно большие, но потом заметно уменьшились, и в основном не превышают 5%, что говорит о хорошей точности. Двухнедельные прогнозы показывают результат лучше, чем четырёхнедельные.

Таблица 1. Ошибки MAPE прогнозирования общего количества заболевших при горизонте прогнозирования две недели

Горизонт прогноза	Индия	Япония	Южная Корея	Италия	Британия	Франция
05.02–18.02	–	–	–	–	–	–
19.02–03.03	–	–	–	–	–	–
04.03–17.03	–	–	102,90	26,74	–	–
18.03–31.03	–	8,24	3,64	5,68	26,01	19,30
01.04–14.04	34,91	5,81	1,94	5,27	13,48	39,64
15.04–28.04	13,81	21,70	0,60	1,11	1,06	26,44
29.04–12.05	1,91	3,83	0,21	0,32	0,94	1,82
13.05–26.05	7,26	0,93	0,13	0,26	1,18	2,21
27.05–09.06	3,94	0,94	1,74	0,32	0,47	2,85
10.06–23.06	2,22	0,64	0,38	0,30	0,72	0,80
24.06–07.07	0,39	2,54	0,37	0,50	0,42	0,28
08.07–21.07	0,55	4,40	1,07	0,25	1,32	0,93
22.07–04.08	0,79	5,00	0,30	0,30	0,30	2,61
05.08–18.08	1,14	5,37	2,07	0,67	0,28	4,97
19.08–01.09	0,42	0,92	2,71	1,56	0,08	1,88
02.09–15.09	0,41	0,66	5,36	0,50	1,32	6,51
16.09–29.09	1,92	0,18	0,49	0,96	0,51	3,46
30.09–13.10	0,84	0,94	0,31	2,76	5,14	3,60
14.10–27.10	0,84	0,93	0,16	1,10	5,03	7,82
28.10–10.11	0,21	1,72	0,39	10,47	1,43	4,38
11.11–24.11	0,29	3,84	3,06	0,82	2,10	0,83
25.11–08.12	0,25	0,71	1,56	0,83	2,00	0,17
09.12–22.12	0,30	1,74	3,70	1,45	3,36	1,42
23.12–31.12	0,06	1,96	1,46	0,25	1,56	0,49

Таблица 2. Ошибки MAPE прогнозирования общего количества заболевших при горизонте прогнозирования четыре недели

Горизонт прогноза	Индия	Япония	Южная Корея	Италия	Британия	Франция
19.02–17.03	–	–	–	–	–	–
18.03–14.04	–	–	0,56	10,39	–	–
15.04–12.05	22,43	279,57	4,12	5,63	9,97	77,37
13.05–09.06	0,70	4,52	2,37	0,33	0,69	1,21
10.06–07.07	1,44	3,00	0,21	0,60	1,30	1,48
08.07–04.08	1,94	16,65	0,66	0,40	2,71	2,53
05.08–01.09	8,22	24,72	9,43	1,70	0,92	11,65
02.09–29.09	3,16	3,96	38,16	1,00	5,33	7,48
30.09–27.10	0,94	1,29	0,25	12,00	3,78	6,32
28.10–24.11	1,80	5,97	2,70	7,13	14,12	6,87
25.11–22.12	0,41	1,19	9,36	4,26	1,97	3,18
23.12–31.12	0,08	1,46	0,83	0,31	0,62	0,50

В таблицах 3, 4 представлены ошибки MAPE прогнозирования количества активных случаев (I) при горизонте прогнозирования две и четыре недели соответственно. Как видно, для активных случаев величина ошибок заметно больше. Двухнедельные прогнозы, как и в случае общего количества заболевших, показывают результат лучше, чем четырёхнедельные.

Таблица 3. Ошибки MAPE прогнозирования количества активных случаев при горизонте прогнозирования две недели

Горизонт прогноза	Индия	Япония	Южная Корея	Италия	Британия	Франция
05.02–18.02	–	–	–	–	–	–
19.02–03.03	–	–	–	–	–	–
04.03–17.03	–	–	117,04	29,56	–	–
18.03–31.03	–	9,50	9,24	6,31	11,22	8,50
01.04–14.04	42,76	5,71	126,43	13,54	2,57	40,29
15.04–28.04	13,80	29,43	2,96	3,54	2,43	45,68
29.04–12.05	10,90	36,03	21,29	3,42	2,58	3,07
13.05–26.05	10,35	13,04	8,96	3,81	1,88	5,57
27.05–09.06	9,18	22,29	22,83	4,67	0,82	5,38
10.06–23.06	12,78	8,83	1,41	7,64	1,01	1,26
24.06–07.07	32,78	11,88	48,82	9,35	0,52	0,33
08.07–21.07	1,89	15,69	25,29	3,02	1,65	1,82
22.07–04.08	6,95	12,04	5,07	4,12	0,38	4,35
05.08–18.08	0,49	36,50	23,81	9,72	0,33	8,06
19.08–01.09	8,34	26,68	8,66	15,43	0,09	3,50
02.09–15.09	1,82	26,06	52,97	6,12	1,56	8,90
16.09–29.09	11,04	31,00	82,13	7,26	0,60	3,56
30.09–13.10	7,05	26,27	10,64	12,08	5,73	3,96
14.10–27.10	4,64	2,55	18,77	1,73	5,51	8,53
28.10–10.11	16,35	7,55	1,92	21,71	1,56	4,30
11.11–24.11	2,83	20,12	23,16	4,51	2,21	0,95
25.11–08.12	5,47	4,04	3,67	6,18	2,15	0,15
09.12–22.12	12,30	5,66	9,52	3,35	3,50	1,50
23.12–31.12	11,27	10,30	10,73	3,47	3,45	1,24

Таблица 4. Ошибки MAPE прогнозирования количества активных случаев при горизонте прогнозирования четыре недели

Горизонт прогноза	Индия	Япония	Южная Корея	Италия	Британия	Франция
19.02–17.03	–	–	–	–	–	–
18.03–14.04	–	–	315,53	19,52	–	–
15.04–12.05	29,23	513,02	4,33	6,98	15,78	149,30
13.05–09.06	12,35	433,80	11,78	5,05	0,35	2,78
10.06–07.07	26,26	17,41	44,53	23,17	1,59	1,84
08.07–04.08	17,51	34,68	93,52	10,18	3,34	4,50
05.08–01.09	4,82	46,61	51,20	12,20	1,07	18,39
02.09–29.09	7,38	4,48	410,25	4,46	6,17	9,13
30.09–27.10	31,32	59,31	6,90	36,49	4,17	7,07
28.10–24.11	38,80	30,72	11,73	17,76	14,68	7,83
25.11–22.12	29,54	8,88	21,80	25,26	2,14	3,03
23.12–31.12	30,58	7,98	13,32	5,71	2,58	1,78

4. Заключение. Интегральная модель притока – оттока, в целом, хорошо зарекомендовала себя при построении двухнедельных прогнозов. Такие прогнозы, как отмечают специалисты национальных органов здравоохранения, могут быть востребованы в процессе планирования деятельности по борьбе с последствиями эпидемий и ресурсном обеспечении противоэпидемиологических мероприятий. Как показали результаты исследования, ориентироваться на четырёхнедельные прогнозы при проведении таких мероприятий нецелесообразно в виду их относительно невысокой точности. В Японии и Южной Корее, как и ожидалось, точность прогнозов оказалась ниже по сравнению с европейскими странами, что связано с возможными особенностями формирования данных в этих странах. Использование степенных и экспоненциальных функций для аппроксимации позволило адаптировать модель к различным типам динамики, что повысило её гибкость. Несмотря на успешное выполнение поставленных задач, выявлены аспекты, требующие дальнейшей доработки. В частности, разработки методов, способных справляться с высокой вариативностью временных рядов. Данное исследование создаёт основу для дальнейшего совершенствования подходов к анализу и прогнозированию эпидемиологических данных, что имеет важное значение для принятия управленческих решений в условиях кризисных ситуаций.

Литература

1. Пономарева П. М. Моделирование динамики распространения COVID-19 на примере первых волн пандемии в Германии // Процессы управления и устойчивость. 2023. Т. 10. № 1. С. 189–194.

2. Radulescu A., Williams C., Cavanagh K. Management strategies in a SEIR-type model of COVID-19 community spread // Scientific Reports. 2020. Vol. 10. P. 1–16.

3. Балыкина Ю. Е., Захаров В. В. Интегральная модель притока и оттока и ее приложения // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2024. Т. 20. № 2. С. 121–135.

4. Захаров В. В., Балыкина Ю. Е. Ретроспективный анализ и прогнозирование распространения вирусов в реальном времени: COVID-19 в Санкт-Петербурге и в Москве в 2020-2021 гг. // Вопросы вирусологии. 2024. Т. 69. № 6. С. 500–508.

5. COVID-19 Data visualization center (internet) // Johns Hopkins coronavirus resource center [Электронный ресурс]: URL: https://coronavirus.jhu.edu/data (дата обращения: 17.10.2024).

УДК 519.172.1, 519.237.8, 519.255

Васильев Е. В.

Тепловые карты, сопряжённые с иерархической структурой медицинских наблюдений

Рекомендовано к публикации старшим преподавателем Ореховым А. В.

1. Введение. Основная задача таксономии – создание системы соподчинённых рангов, которая служит для построения естественной классификации. Математически таксономией является древообразная иерархическая структура классификаций определённого набора объектов. Вверху этой структуры находится корневой таксон, который относится ко всем объектам данной таксономии. На различных уровнях иерархии между основанием и вершиной находятся таксоны, каждый из которых подчинён одному и только одному таксону более высокого ранга [1, 2].

Тепловая карта (англ. heatmap) – это графическое представление данных, где их значения в таблице отображаются при помощи цвета. Этот метод позволяет наглядно представлять результаты экспериментов и выявлять скрытые закономерности. В медицине и молекулярной биологии они могут использоваться для представления данных по экспрессии генов в различных образцах. Тепловые карты строятся в виде таблицы, где цвет прямоугольника показывает уровень экспрессии, а столбцы и строки представляют различные гены или образцы.

Для увеличения наглядности полученных результатов используется сопряжение тепловой карты с иерархической структурой генов и образцов, построенной при помощи иерархической агломеративной кластеризации [3]. Результат кластеризации представляется в виде дендрограммы – древообразной структуры, где каждая ветвь соответствует группе элементов. Ребра дендрограммы соединяют элементы, которые находятся ближе друг к другу в пространстве признаков [4, 5]. Сами дендрограммы располагаются на полях, сверху, снизу, слева или справа от сопряжённой тепловой карты.

Васильев Егор Владимирович – студент, Санкт-Петербургский государственный университет; e-mail: st097569@student.spbu.ru, тел.: +7(982)310-83-54

Если элементы выборочной совокупности упорядочены в соответствии с результатами иерархической агломеративной кластеризации, то на тепловой карте они расположены рядом и имеют похожие профили выраженности генов или других характеристик. Этот подход помогает строить паттерны (устойчивую комбинацию результатов исследований) и корреляции на больших наборах данных, делая сопряжённые с иерархической структурой тепловые карты эффективным инструментом для анализа сложных биологических систем и медицинских наблюдений.

2. Вычислительный эксперимент. В качестве набора экспериментальных данных использовались нормализованные экспрессии раково-тестикулярных генов (РТГ) онкологических больных. В опухолях различных типов наблюдается неодинаковая частота экспрессии РТГ, которые принадлежат несовпадающим мультигенным семействам и чаще всего локализованы в Х-хромосоме.

Для проведения вычислительных экспериментов использовался язык программирования Python 3.7 с библиотеками NumPy, SciPy, matplotlib и seaborn. Также применялись оболочка Visual Studio Code и среда свободного доступа Colaboratory для работы с Jupyter Notebook от Google. Особое внимание стоит уделить методу scipy.cluster.hierarchy.linkage, результаты которого были использованы для определения точки остановки кластеризации и построения дендрограммы. В метод передавался параметр method со значением “ward”, при этом остальные параметры задавались по умолчанию. Для построения дендрограмм использовался метод scipy.cluster.hierarchy.dendrogram, а для визуализации тепловых карт применялась функция seaborn.heatmap.

Момент остановки определялся при помощи аппроксимационно-оценочных критериев [6, 7]. Методика построения иерархии вложенных кластеров описана в статье [8]. В качестве количественной меры устойчивости результатов кластеризации использовалась величина отрезка $Q_i = [\alpha_i, \beta_i]$ изменения коэффициента тренда $q \in [\alpha_i, \beta_i]$, при котором для выборочной совокупности X получается один и тот же результат. Коэффициент тренда – это параметр, введённый с дополнительным преобразованием множества минимальных расстояний для изменения чувствительности аппроксимационно-оценочных критериев. Результат типологизации при определённом отрезке Q_i называется промежутком устойчивой кластеризации.

На рис. 1 представлен первый промежуток устойчивой кластеризации, на основе которого можно наблюдать чёткую разделимость профилей экспрессий РТГ, что подтверждается соответствующей тепловой картой.

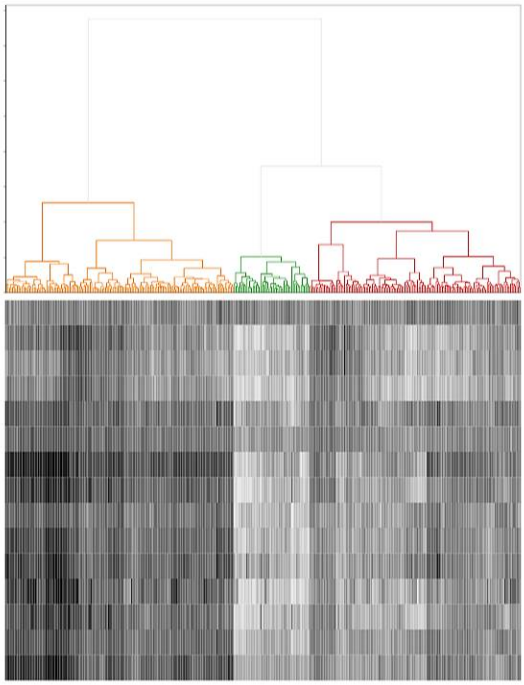


Рис. 1. Иерархия кластеров и сопряжённая тепловая карта. Первый промежуток устойчивой кластеризации

При рассмотрении второго промежутка устойчивой кластеризации на рис. 2, можно отметить, что на данном этапе происходит дальнейшее разделение данных. Формируются более мелкие кластеры, которые, несмотря на свой меньший размер, также чётко разделяют профили экспрессий. Однако важно подчеркнуть, что эти более мелкие кластеры можно рассматривать как подкластеры кластеров, полученных на первом промежутке устойчивой кластеризации.

Таким образом, наблюдается процесс детализации результатов типологизации, где каждый новый уровень уточняет и подразделяет группы наблюдений.

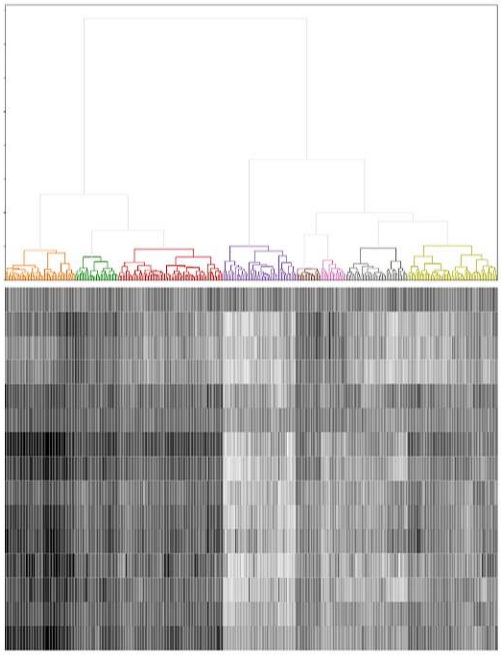


Рис. 2. Иерархия кластеров и сопряжённая тепловая карта.
Второй промежуток устойчивой кластеризации

3. Заключение. Полученные результаты иллюстрируют, как при помощи аппроксимационно-оценочных критериев, являющихся аналитическим обобщением эвристического «метода локтя» [9], можно эффективно определять моменты остановки процесса кластеризации. Эмпирическая оценка результатов при помощи тепловых карт, сопряжённых с их иерархической структурой, подтверждает высокую степень адекватности и точности данного метода, что позволяет использовать его для анализа медицинских наблюдений и при построении моделей, например, в задачах математической иммунологии [10, 11].

Литература

1. Джеффри Ч. Биологическая номенклатура / Пер. с англ. М.: Мир, 1980. 119 с.
2. Зуев В. В., Розова С. С. Проблема способа бытия таксона в биологической таксономии // Вопросы философии. 2003. № 2. С. 90–103.
3. Wilkinson L., Friendly M. The history of the cluster heat map // The American Statistician. 2009. Vol. 63. No 2. P. 179–184.
4. Жамбю М. Иерархический кластер-анализ и соответствия. М.: Финансы и статистика, 1988. 345 с.
5. Calirnski T., Harabasz J. A dendrite method for cluster analysis // Communications in Statistics. 1974. No 3. P. 1–27.
6. Orekhov A. V. Agglomerative method for texts clustering // Proceedings of the 5th International Conference on Internet Science (INSCI 2018) / Ed. by S. Bodrunova. Switzerland. Springer: Cham, 2019. Vol. 11551. P. 19–32.
7. Orekhov A. V. Quasi-deterministic processes with monotonic trajectories and unsupervised machine learning // Mathematics. 2021. Vol. 9. Art. no 2301.
8. Орехов А. В., Васильев Е. В. Бинарные метрические деревья и иерархия вложенных кластеров // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2024. Т. 20. № 4. С. 487–499.
9. Thorndike R. L. Who belongs in the family? // Psychometrika. 1953. Vol. 18. P. 267–276.
10. Кузнецов С. Р. Математическая модель иммунного ответа // Вестник Санкт-Петербургского университета. Серия 10: Прикладная математика. Информатика. Процессы управления. 2015. Т. 11. № 4. С. 72–87.
11. Кузнецов С. Р., Лыкосов В. М., Орехов А. В., Шишкин В. И. Модель пролиферации и дифференцировки неоднородной клеточной популяции // В сборнике: Математическое и компьютерное моделирование в биологии и химии. III Международная научная Интернет-конференция: Материалы конференции. 2014. С. 95–103.

УДК 51-76

Давыденко Б. Н.

Текстурный подход к анализу микрофотографий клеточных культур

Рекомендовано к публикации профессором Котиной Е. Д.

Актуальность. Задача оценки количества клеток на изображении возникает в цитологических исследованиях, как один из методов подсчета клеток. Большинство методов подсчета основаны на получении суспензии клеток [1], однако при культивации клеток в монослое может требоваться неоднократная оценка их количества [2, 3].

В настоящее время при решении задач оценки числа клеток разных типов на изображениях используются методы автоматической сегментации [4, 5]. Текстурный подход к анализу микрофотографий клеток применялся ранее при изучении миграции [6].

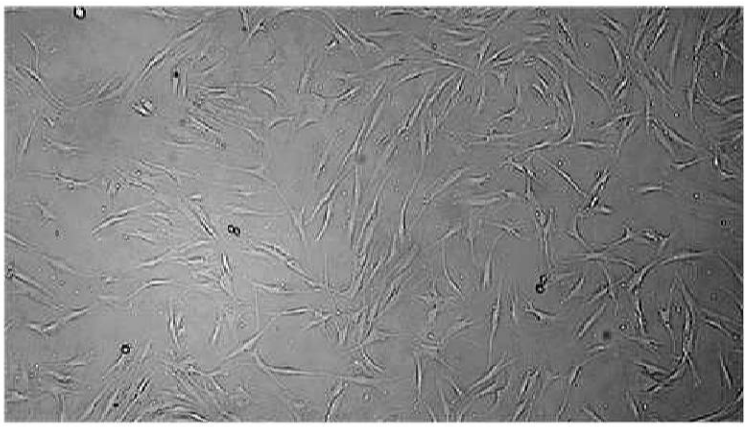


Рис. 1. Пример микрофотографии

Данные. Для получения изображений использовались оптический микроскоп Nikon Eclipse TS100 и цифровая камера XCAM

Давыденко Богдан Николаевич – магистрант, Санкт-Петербургский государственный университет; e-mail: st086788@student.spbu.ru, тел.: +7(900)569-15-87

Family 1080P. Были получены 10 микрофотографий культуры дермальных фибробластов линии DF-1 [7]. Изображения (см. рис. 1) в формате .tif монохромны и имеют размер 1920×1080 пикселей, 256 уровней интенсивности. Для каждого из изображений было подсчитано количество клеток. Изображения предоставлены сотрудниками Центра клеточных технологий ИНЦ РАН.

Методы. При анализе для каждой микрофотографии рассчитывались текстурные дескрипторы [8], вычисляемые на основании гистограммы яркости и матрицы Харалика. Для исследования полученных характеристик применялись методы регрессионного анализа [9]. Исследование корреляций показало, что характеристик гистограммы не достаточно для решения поставленной задачи. Результат анализа дескрипторов, характеризующих матрицу Харалика, представлен в таблице.

Таблица. Характеристики матрицы Харалика

Дескриптор	Коэффициент корреляции	Средняя абсолютная ошибка, %
Однородность	−0,9788	4,2
Энтропия	0,8143	10
Энергия	−0,8659	8,3
Максимум вероятности	−0,8009	10,7
Корреляция	−0,1689	20,8
Контраст	0,9069	7,1

Наиболее эффективным дескриптором для построения оценки количества клеток на микрофотографии является однородность:

$$U = \sum_{i=0}^{K-1} \sum_{j=0}^{K-1} \frac{p_{ij}}{1 + |i - j|},$$

где K – количество уровней интенсивности, p_{ij} – частота смежности пикселей с интенсивностями i и j. На рис. 2 представлены линия регрессии и доверительные интервалы для линии регрессии и значений зависимой переменной для этого дескриптора.

Выводы. Для решения задачи оценки количества клеток на микрофотографии применима модель линейной регрессии по характеристике однородности. Это подтверждается высоким коэффициентом корреляции и приведенными доверительными интервалами.

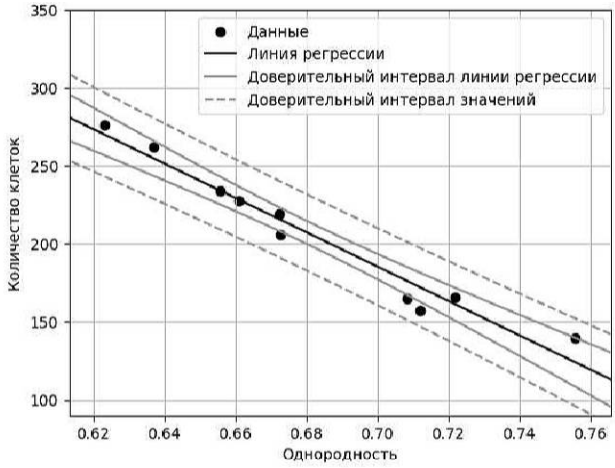


Рис. 2. График линии регрессии

Предложенная методика не требует алгоритмически сложных вычислений. Для более широкого применения, с использованием другого оборудования или иного типа клеток, достаточно провести анализ изображений с предварительно подсчитанным количеством клеток. Метод не требует окрашивания клеток или получения суспензии, что позволит применять его в прижизненных исследованиях клеточных культур.

Литература

1. ОФС.1.7.2.0008.15 // Определение концентрации микробных клеток [Электронный ресурс]: URL:https://pharmacopoeia.regmed.ru/pharmacopoeia-projects/izdanie-14/1/1-7/1-7-2/?vers=2341/ (дата обращения: 14.04.2025).

2. Халтобина М. М., Елизарова М. В. Анализ выживаемости клеток в рамках RMR-модели // Процессы управления и устойчивость. 2024. Т. 11. № 1. С. 155–162.

3. Токин И. Б., Толстая М. В., Филимонова Г. Ф. Кинетика клеточной популяции кишечного эпителия // Вестник Санкт-Петербург-

ского университета. Серия 10: Прикладная математика. Информатика. Процессы управления. 2006. № 1. С. 84–93.

4. Дырнаев А. В. Метод подсчета эритроцитов на изображениях мазков крови // Научно-технический вестник информационных технологий, механики и оптики. 2011. № 6(76). С. 17–22.

5. Тарков М. С., Осипов М. И. Оценивание числа клеток на изображениях цитологических растительных препаратов // Известия Томского политехнического университета. Инжиниринг георесурсов. 2013. № 5. С. 187–193.

6. Qin J., Chittenden T. W., Gao L., Pearlman J. D. Automated migration analysis based on cell texture: method & reliability // BMC Cell Biology. 2005. No 6. P. 1–6.

7. DF-1 // Центр коллективного пользования «Коллекция культур клеток позвоночных» Института цитологии [Электронный ресурс]: URL:https://incras-ckp.ru/catalog/df-1/ (дата обращения: 10.03.2025).

8. Гонсалес Р., Вудс Р. Цифровая обработка изображений. 3 изд. М.: Техносфера, 2012. 1104 с.

9. Красько О. В. Статистический анализ данных в медицинских исследованиях. Минск: МГЭУ им. А. Д. Сахарова, 2014. 317 с.

УДК 004.891.3

Кириченко Е. Ю., Громов Р. М., Блеканов И. С., Назаров В. Д., Денисов Д. Г., Лапин С. В.

Методы диагностики анемии в лабораторных исследованиях на основе машинного обучения

1. Введение. Машинное обучение открывает новые возможности для анализа сложных и многомерных данных в различных областях. Сфера лабораторных исследований не является исключением. Существуют AI-технологии и инструменты, направленные на то, чтобы автоматизировать процессы принятия решений, поиск в данных анализов аномалий, которые не видны врачу. Так, например, в [1] авторы с помощью модели на основе метода машины опорных векторов прогнозируют начало сепсиса с точностью в 92,5%. В [2] команда ученых применила методы машинного обучения для анализа биомаркеров в крови с целью раннего выявления рака. В [3] рассматривается подход к построению стратегии автоматического выделения лимфоцитов, моноцитов и гранулоцитов при помощи модифицированного агломеративного центроидного метода кластеризации. В [4] поднимается проблема оценки саркопении и возможностей применения машинного обучения к задаче сегментации мышечных групп.

В данной работе предлагается решение задачи исследования методов предварительной обработки данных и машинного обучения

Кириченко Елизавета Юрьевна – студент, Санкт-Петербургский государственный университет; e-mail: st087544@student.spbu.ru, тел.: +7(914)335-73-15

Громов Рамиль Махирович – магистрант, Санкт-Петербургский государственный университет; e-mail: st087271@student.spbu.ru, тел.: +7(980)737-43-64

Блеканов Иван Станиславович – доцент, Санкт-Петербургский государственный университет; e-mail: i.blekanov@spbu.ru, тел.: +7(812)363-65-37

Назаров Владимир Дмитриевич – заведующий лабораторией молекулярной диагностики НМЦ Минздрава России по молекулярной медицине, ПСПбГМУ имени акад. И.П.Павлова; e-mail: autoimmun@mail.ru, тел.: +7(812)994-53-24

Денисов Дмитрий Геннадьевич – медицинский директор, лабораторная служба «ХЕЛИКС»; e-mail: ddenisov@helix.ru, тел.: +7(812)607-66-07

Лапин Сергей Владимирович – заведующий лабораторией диагностики аутоиммунных заболеваний НМЦ Минздрава России по молекулярной медицине, ПСПбГМУ имени акад. И.П.Павлова; e-mail: autoimmun@mail.ru, тел.: +7(812)994-53–24

для качественного установления вида анемии на реальных лабораторных исследованиях. Решение этой задачи является сложным вопросом, поскольку существующие научные работы [5–7] отмечают проблемы дороговизны проведения некоторых необходимых анализов, сосредоточенности на одном типе анемии, дисбаланса классов, недостатка открытых данных с большим количеством проведенных тестов.

2. Предлагаемое решение. В данной работе предлагается следующая схема обработки и анализа лабораторных исследований для выявления у пациентов конкретного класса анемии (см. рис. 1).

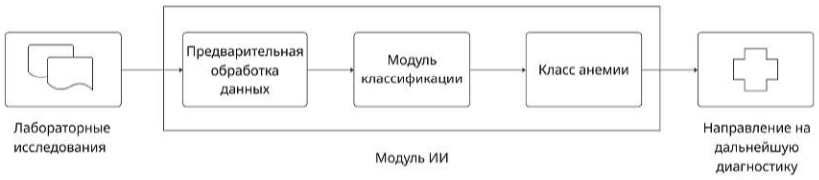


Рис. 1. Схема данных

Схема на рис. 1 состоит из следующих этапов:

1. Лабораторные исследования. На вход системе поступает информация об общем анализе крови, поле и возрасте пациентов.

2. Предварительная обработка данных. Блок содержит в себе следующие этапы: удаление пропусков и обработка ошибок в данных, стандартизация, дискретизация и бинаризация признаков, создание полиномиальных признаков.
3. Модуль классификации. В данном модуле реализуется четыре наиболее эффективных методов многоклассовой классификации: на основе машины опорных векторов, логистической регрессии, случайного леса, градиентного бустинга над решающими деревьями. В текущей реализации классификатор может выявлять один из следующих пяти классов: условно здоровые пациенты, B12-анемия, железодефицитная анемия, фолиеводефицитная анемия, талассемия.

4. Класс анемии. Возвращает один из пяти вышеперечисленных классов.
5. Направление на диагностику. На основе предсказания модели система возвращает результат – рекомендацию, какой тест пациенту пройти следующим.

Предложенная система была реализована с помощью специализированных библиотек языка Python [8, 9].

3. Набор лабораторных данных. Медицинскими экспертами были предоставлены наборы данных по следующим видам анемий: данные по тесту ферритина, витамина B12, фолатов, талассемии, информация об условно здоровых пациентах, и другим редким видам анемий. Дополнительно поступила таблица с условно здоровыми пациентами. Полученные данные были стандартизированы и приведены к единому формату.

В итоге тестовая выборка данных состояла из 44099 пациентов, называемых условно здоровыми, 58 пациентов с B12-анемией, 4392 пациентов с железодефицитной анемией, 13 пациентов с фолиеводефицитной анемией и 100 пациентов с талассемией. Данная выборка разделена на обучающую и тестовую в соотношении 70/30.

4. Эксперимент. Далее рассмотрим один из проведенных в исследовании опытов.

4.1. Постановка эксперимента. В работе был поставлен эксперимент по сравнительному анализу и выбору лучшего метода классификации в разработанном решении из пункта 2: машина опорных векторов, логистическая регрессия, случайный лес, градиентый бустинг над решающими деревьями. Требовалось обучить классификаторы из пункта 2 и вычислить метрики оценки качества модели: точность, полноту, F1-меру, и на их основе выбрать лучший классификатор.

4.2. Результаты эксперимента. Оценка качества классификаторов представлена в таблице 1. Детализированная оценка метода логистической регрессии приведена в таблице 2.

Таблица 1. Оценка качества классификаторов

Классификатор	**Точность**	**Полнота**	**F1-мера**
Машина опорных векторов	0,77	0,78	0,77
Логистическая регрессия	0,81	0,81	0,81
Случайный лес	0,71	0,72	0,71
Градиентный бустинг над решающими деревьями	0,69	0,68	0,68

Таблица 2. Оценка качества классификатора логистической регрессии

Класс	**Точность**	**Полнота**	**F1-мера**
Условно здоровые	1,00	1,00	1,00
B12-анемия	0,87	0,76	0,81
Железодефицитная анемия	0,99	1,00	0,99
Фолиеводефицитная анемия	0,40	0,50	0,44
Талассемия	0,79	0,77	0,78
Среднее	0,81	0,81	0,81

5. Заключение. Наиболее успешным оказался классификатор логистической регрессии. Найденное решение может быть оценено как хорошее: его точность по ключевым метрикам составляет 81%, оно легко интерпретируемо за счет выбора подходов, а также с приемлемой точностью по одному лишь общему анализу крови, полу и возрасту позволяет заподозрить у пациента редкую генетическую аномалию – талассемию. В дальнейшем планируется продолжить работу по улучшению качества модели с помощью расширения блока предварительной обработки данных и более детальной работой над проблемой дисбаланса классов.

Литература

1. Rasha M. Abd El-Aziz, Alanazi R. Early detection of sepsis using machine learning algorithms // Alexandria Engineering Journal. 2025. Vol. 111. P. 47–56.

2. Xie Y., Meng W. Y., Li R. Z. et al. Early lung cancer diagnostic biomarker discovery by machine learning methods // Translational Oncology. 2021. Vol. 14. No 1. P. 1–10.

3. Орехов А. В., Шишкин В. И., Кудрявцева Г. В., Павилайнен Г. В., Шишкин В. В., Людкевич Н. С. Автоматическое получение «упрощенной лейкоцитарной формулы» методами проточной цитофлуориметрии и кластерного анализа // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2023. Т. 19. № 4. С. 469–483.

4. Шмидт Я. Н. Анализ саркопении методами машинного обучения // Процессы управления и устойчивость. 2023. Т. 10. № 1. С. 211–215.

5. Karagul Yildiz T., Yurtay N., Onec B. Classifying anemia types using artificial learning methods // Engineering Science and Technology, an International Journal. 2021. Vol. 24. P. 50–70.

6. Kilicarslan S., Celik M., Sahin S. Hybrid models based on genetic algorithm and deep learning algorithms for nutritional Anemia disease classification // Biomedical Signal Processing and Control. 2021. Vol. 63. P. 1–10.

7. Uvaliyeva I., Belginova S., Borozenets D. Analysis of the machine learning methods effectiveness for morphological classification of anemia // 2023 7th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT). 2023. P. 1–5.

8. Pandas [Электронный ресурс]: `URL:https://pandas.pydata.org/` (дата обращения: 12.03.25).

9. Scikit-learn [Электронный ресурс]: `URL:https://scikit-learn.org/stable/` (дата обращения: 12.03.25).

УДК 004.89

Козин А. А.

Классификация новообразований молочной железы по BI-RADS с помощью нейронных сетей

Рекомендовано к публикации доцентом Гончаровой А. Б.

1. Введение. Рак молочной железы (РМЖ) занимает одно из ведущих мест по распространенности среди онкологических заболеваний у женщин [1]. Для стандартизации диагностического процесса и точной оценки вероятности злокачественности опухолей была разработана система BI-RADS (Breast Imaging Reporting and Data System), которая помогает врачам оценивать риск злокачественности опухолей, основываясь на специфических характеристиках, и предлагает последующие действия в зависимости от уровня подозрений [2, 3].

Основная цель исследования – разработка моделей машинного обучения для автоматического предсказания категории BI-RADS на основе данных обследований.

2. Данные. Для исследования были использованы данные ультразвуковых исследований, предоставленные НМИЦ онкологии им. Н. Н. Петрова. В выборку вошли более 30 признаков, охватывающих различные параметры пациентов и характеристики новообразований, что создает основу для построения прогностических моделей [4].

Для повышения качества моделей был проведен предварительный анализ данных, который включал несколько ключевых этапов [5]:

1. Исключены случаи, где врачебный диагноз не совпадал с данными гистологии. Это позволило повысить точность предсказаний, оставив в выборке только данные с подтвержденным диагнозом.
2. Удалены признаки, содержащие уже готовые диагностические заключения. Это помогло избежать избыточных данных и сосредоточить модель на анализе исходных характеристик.

Козин Артем Андреевич – студент, Санкт-Петербургский государственный университет; e-mail: st097212@student.spbu.ru, тел.: +7(913)292-40-41

3. Устранена мультиколлинеарность среди признаков, которая могла бы искажать результаты моделей. Для анализа мультиколлинеарности использовался коэффициент (VIF). Признаки с высокими значениями VIF были удалены, что позволило сделать модель более стабильной и устойчивой к переобучению.

После обработки данных в выборке остались следующие признаки: возраст, менструальный статус, мутация гена BRCA 1,2, размер УЗИ, капсула, четкость, ровность контура, форма, ориентация образования, ЦДК васкуляризация, ЦДК локализация кровотока, УЗИ количество очагов РМЖ, StR, CEUS паттерн, CEUS тип кривой, $C_{\max}$, $t_{\max}$.

3. Построение моделей. Для классификации были применены три модели машинного обучения [6], каждая из которых имеет свои технические особенности и преимущества:

1. Линейная регрессия. Так как BI-RADS представлен дискретными классами, предсказанные значения округлялись до ближайшего класса.
2. Логистическая регрессия. Использовалась softmax-регрессия – многоклассовая модификация логистической регрессии.
3. Случайный лес. Гиперпараметры подобраны поиском по сетке.

Для оценки моделей данные были разделены на тренировочные и тестовые выборки с помощью стратифицированного разбиения, что позволило сохранить пропорции классов.

Каждая модель оценена по ключевым метрикам: точности, F1-мере и ROC-AUC, что позволило объективно сравнить их работу [6].

Таблица 1. Сравнение результатов для метрики precision

	Линейная регрессия	Логистическая регрессия	Случайный лес
Class 2	0,97	0,90	0,95
Class 3	0,40	0,50	0,54
Class 5	1,00	1,00	1,00

Таблица 2. Сравнение результатов для метрики recall

	Линейная регрессия	Логистическая регрессия	Случайный лес
Class 2	0,81	0,94	0,89
Class 3	0,89	0,44	0,78
Class 5	0,86	0,95	0,95

Таблица 3. Сравнение результатов для F1-меры

	Линейная регрессия	Логистическая регрессия	Случайный лес
Class 2	0,88	0,92	0,92
Class 3	0,55	0,47	0,64
Class 5	0,93	0,98	0,98

Таблица 4. Сравнение результатов для метрики ROC-AUC

Линейная регрессия	Логистическая регрессия	Случайный лес
0,9142	0,9036	0,9507

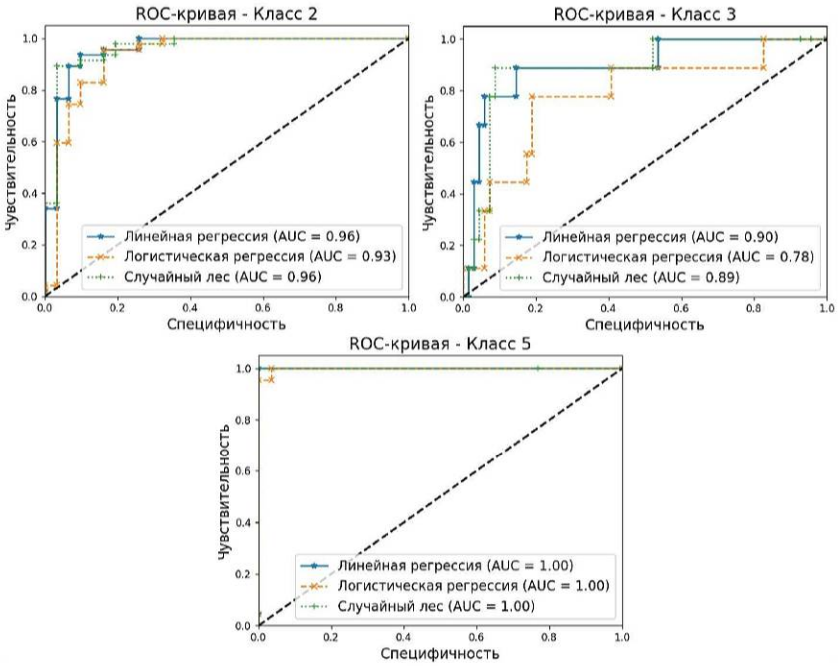


Рис. 1. Графики ROC-кривых для трех моделей по трем классам

Сравнение трех моделей – линейной регрессии, логистической регрессии и случайного леса – показало различия в их способности классифицировать категории BI-RADS (см. таблицы 1–4 и рис. 1):

1. Логистическая регрессия продемонстрировала умеренную точность. Классы 2 и 5 были классифицированы достаточно точно. Однако для класса 3 метрики оказались низкими.
2. Линейная регрессия показала улучшенную точность по сравнению с логистической регрессией. Метрики для классов 2 и 5 также были высокими, а для класса 3 показатели улучшились. Этот метод может быть использован для начального этапа анализа данных или в случаях, где требуется модель с понятной интерпретацией.
3. Случайный лес продемонстрировал наилучшие результаты. Метрики для классов 2 и 5 были очень высокими. Для класса 3 показатели также были выше, чем у других моделей. Это делает модель оптимальным выбором для задач автоматической диагностики.

В итоге, проведенный анализ трех моделей классификации показал разнообразие результатов, с наилучшими показателями у модели случайного леса. Несмотря на достигнутые успехи, особенно в точности для классов 2 и 5 по BI-RADS, остается значительный потенциал для улучшения классификации редких классов, таких как класс 3.

Эти результаты подчеркивают необходимость дальнейших исследований и оптимизации моделей, в особенности для классификации категории BI-RADS 3, что откроет новые возможности для повышения точности и надежности в медицинской диагностике.

Литература

1. Valieva N. L., Goncharova A. B. Breast cancer classification in mammograms // Процессы управления и устойчивость. 2024. Т. 11. № 1. С. 168–172.

2. Семиглазов В. В., Топузов Э. Э. Рак молочной железы. М.: Медпресс-информ, 2009. 172 с.

3. Бусько Е. А., Гончарова А. Б и др. Модель принятия диагностических решений на основе мультипараметрических ультразвуковых показателей образований молочной железы // Вопросы онкологии. 2020. Т. 66. № 6. С. 653–658.

4. Бусько Е. А. Мультипараметрическое ультразвуковое исследование с применением соноэластографии и эхоконтрастирования у пациенток с образованиями молочной железы. Патент RU 2019620435, 19.03.2019.

5. Гончарова А. Б., Бусько Е. А. Программная реализация системы принятия диагностических решений на основе мультипараметрических ультразвуковых показателей образований молочной железы // Сибирский журнал клинической и экспериментальной медицины. 2020. Т. 35. № 4. С. 137–142.

6. Dolgintseva E., Wu H., Petrosian O. et al. Comparison of multi-step forecasting methods for renewable energy // Energy Systems. 2024. P. 1–32. https://doi.org/10.1007/s12667-024-00656-w

УДК 004.8

Кривошеин А. Е., Хоменко И. Е.

Многоклассовый и двухэтапный подход к классификации и обработке медицинских данных

Рекомендовано к публикации доцентом Гончаровой А. Б.

1. Введение. Сон является одной из важнейших составляющих здоровья. Нарушения сна значительно ухудшают качество жизни человека и приводят к серьезным последствиям в долгосрочной перспективе. Самыми распространенными болезнями сна являются бессонница и апноэ.

Бессонница характеризуется непосредственно отсутствием сна, может быть вызвана рядом факторов как соматического, так и психологического характера [1]. Апноэ же характеризуется кратковременным прерыванием дыхания во время сна, что может приводить к серьезному дефициту кислорода [2].

Целью работы является сравнение основных подходов к построению классификационной модели при работе с несколькими классами на примере задачи классификации расстройств сна. Сформировать рекомендации по нормализации сна.

Для достижения поставленной цели необходимо решить следующие задачи:

- провести предварительную обработку данных, включая заполнение пропущенных значений и кодирование категориальных признаков;
- разделить данные на обучающую и тестовую выборки для дальнейшей оценки качества моделей;
- построить модель многоклассовой классификации, способную сразу разделять пациентов на здоровых, с бессонницей и с апноэ, оценить качество модели;

Кривошеин Алексей Евгеньевич – студент, Санкт-Петербургский государственный университет; e-mail: st117264@student.spbu.ru, тел.: +7(913)803-17-04

Хоменко Ирина Евгеньевна – магистрант, Санкт-Петербургский государственный университет; e-mail: st087797@student.spbu.ru, тел.: +7(924)150-11-84

- построить модели последовательной классификации, предварительно разделив на здоровых и больных, а дальше классифицировать тип расстройства сна, оценить качество моделей;

- сравнить качество моделей с помощью метрик верности (accuracy), точности (precision), чувствительности (recall) и F1-меры, выявить преимущества и недостатки подходов на примере данной задачи;
- выявить основные признаки, влияющие на появление расстройств сна, для уменьшения вероятности их возникновения.

2. Материалы. Для исследования использовались открытые данные из базы данных Kaggle [3], которая содержит наблюдения за образом жизни и здоровьем людей различных профессий. Данные, представленные в файле, содержат информацию о 374 пациентах, включая их пол, возраст, профессию, продолжительность сна, качество сна, уровень физической активности, уровень стресса, индекс массы тела (ИМТ), артериальное давление, частоту сердечных сокращений, количество ежедневных шагов, в качестве результирующего параметра выступает показатель наличия расстройств сна.

Расстройства сна разделены на категории: бессонница, апноэ, а также отсутствие патологий сна. Для проверки качества работы модели исходный набор данных разбивается на обучающую выборку, содержащую 75% данных, и тестовую для обучения и верификации построенной модели соответственно.

3. Методы. Для классификации болезней сна были выбраны два подхода [4]:

- многоклассовая классификация: модель сразу предсказывает один из трех классов (здоровые, бессонница, апноэ);
- двухэтапная классификация: модель сначала определяет, есть ли у человека расстройство сна, а затем, если оно есть, предсказывает конкретное заболевание (бессонница или апноэ).

Для сравнения эффективности двух подходов будут использоваться метрики [5].

4. Многоклассовая классификация. Данная классификация предсказывает один из трех классов непосредственно. В работе

используется алгоритм случайного леса [6]. В результате применения данного подхода была получена модель с верностью, равной 0,89, и характеристиками, представленными в таблице 1.

Таблица 1. Многоклассовая классификация

№	Точность	Чувствительность	F1-мера
Бессонница	0,83	0,79	0,81
Здоровые	0,97	0,93	0,95
Апноэ	0,72	0,87	0,79

Модель учитывает все взаимосвязи между признаками и классами сразу, что минимизирует накопление ошибки. Однако несбалансированность классов может вести к склонности модели к ошибкам.

Модель показала наибольшую зависимость от признаков ИМТ, артериальное давление, продолжительность сна, возраст.

5. Двухэтапная классификация. Данная классификация использует две модели: первая модель определяет, есть ли у человека расстройство сна, а вторая предсказывает конкретное заболевание.

Первая модель строилась на основе алгоритма из семейства деревьев решений (DecisionTreeClassifier), а вторая модель – методом случайного леса.

Первая модель при тестировании на аналогичной тестовой выборке показала верность, равную 0,95, а вторая – 0,91, остальные метрики приведены в таблице 2, 3.

Таблица 2. Качество модели, определяющей здоров ли пациент

№	Точность	Чувствительность	F1-мера
Здоровые	0,98	0,93	0,96
Больные	0,89	0,97	0,93

Таблица 3. Качество модели, определяющей вид расстройства сна

№	Точность	Чувствительность	F1-мера
Бессонница	0,94	0,89	0,92
Апноэ	0,88	0,93	0,90

Для сравнения с многоклассовым подходом, найдем показатели качества двухэтапной модели в целом. Модель показала верность 0,92, остальные результаты приведены в таблице 4.

Таблица 4. Многоклассовая классификация

№	Точность	Чувствительность	F1-мера
Бессонница	0,85	0,89	0,87
Здоровые	0,98	0,93	0,96
Апноэ	0,82	0,93	0,87

Происходит улучшение качества классификации при помощи упрощения задач, так как каждая модель решает более простую задачу, что повышает точность предсказаний. Также качество повышается за счет возможности использования разных моделей для разных этапов, что позволяет выбрать наиболее подходящие модели для каждой задачи.

Однако необходимо учесть увеличение сложности, так как необходимо обучать и поддерживать две модели вместо одной. Кроме того, возникает риск накопления ошибок, в связи с тем, что ошибка на первом этапе приводит к неправильному предсказанию на втором этапе.

Найдем основные признаки, на которых строятся модели. Модель разделения пациентов на здоровых и больных базируется на таких признаках, как ИМТ (а именно, избыточный вес), частота сердечных сокращений, возраст, артериальное давление. А модель, классифицирующая расстройства сна, строится на данных о физической активности, кровяном давлении, возрасте и количестве ежедневных шагов.

Из представленных данных видно, что важным фактором, влияющим на возникновение расстройств сна, является излишний вес, а вот диагностирование проблем со сном базируется на таких факторах, как физическая активность, кровяное давление, возраст.

6. Заключение. В результате исследования были построены классификационные модели высокой точности и показано, что в задачах медицинской диагностики в случае возможности логического разделения на внешнюю и внутреннюю модель целесообразно использовать двухэтапную классификацию.

Многоклассовая классификация более эффективна, если данные сбалансированы и классы хорошо разделимы. Двухэтапная классификация более эффективна, если данные несбалансированы, и если задача классификации на внешнем этапе проще, чем на внутреннем, как в представленном случае.

В результате выбор подхода зависит от конкретных характеристик данных и требований к модели, а часто может сказываться на результате классификации и вовсе несущественно. В любом случае, важно тщательно оценивать качество модели и выбирать наиболее подходящий метод для решения задачи.

Кроме того, в результате исследования было выявлено, что для

нормализации качества сна необходимо следить за индексом массы тела.

Литература

1. Ohayon M. M. Epidemiology of insomnia: what we know and what we still need to learn // Sleep Medicine Reviews. 2002. Vol. 6. No 2. P. 97–111.

2. Buysse D. J., Reynolds C. F., Monk T. H., Kupfer D. J., Berman S. R. The pittsburgh sleep quality index: a new instrument for psychiatric practice and research // Psychiatry Research. 1989. Vol. 28. No 2. P. 193–213.

3. Sleep health and lifestyle dataset [Электронный ресурс]: `URL:https://www.kaggle.com/datasets/uom190346a/sleep-health-and-lifestyle-dataset` (дата обращения: 13.03.25).

4. Kotsiantis S. B., Zaharakis I., Pintelas P. Supervised machine learning: a review of classification techniques // Emerging Artificial Intelligence Applications in Computer Engineering. 2007. P. 3–24.

5. Lundberg S. M., Lee S. I. A unified approach to interpreting model predictions // Advances in Neural Information Processing Systems. 2017. Vol. 30. P. 4765–4774.

6. Breiman L. Random forests // Machine Learning. 2001. Vol. 45. No 1. P. 5–32.

УДК 004.896

Моргач Д. Ю.

Фильтрация электромиографических сигналов с целью классификации действий

Рекомендовано к публикации доцентом Гончаровой А. Б.

1. Введение. Электромиографические (ЭМГ) сигналы – это электрическая активность, возникающая в мышцах при их сокращении. В дальнейшем анализ данной активности может быть использован для назначения команд для управления устройствами [1, 2].

2. Актуальность. В современном мире бионический протез представляет собой высокотехнологическое решение для восстановления утраченных конечностей или функций организма. Целью работы является анализ электромиографических сигналов и их разметки. Данный метод может применяться для создания бионических протезов [3, 4].

3. Сравнение с бионическими аналогами. На основе датчиков ЭМГ и платы Arduino планируется создание устройства при существовании бионических протезов. Основными преимуществами являются стоимость, гибкость, практическое применение. Низкая стоимость обусловлена дешевизной компонентов. Гибкость состоит в возможности постоянной модификации или улучшения посредством добавления новых функций: увеличение ёмкости батареи, усовершенствование алгоритмов обработки сигналов, повышение точности работы приводов и назначение своих уникальных команд управления. Протез может использоваться для ежедневной жизни в выполнении базовых задач [5].

4. Регистрация сигналов. Для регистрации ЭМГ сигналов используются электроды, которые могут быть поверхностными или внутренними (инвазивными), имлпантируемыми в мышцы. Электроды регистрируют потенциалы действия мышц, возникающие при их сокращении. Эти сигналы могут варьироваться по амплитуде, частоте и продолжительности в зависимости от усилия и типа движения.

Моргач Дмитрий Юрьевич – магистрант, Санкт-Петербургский государственный университет; e-mail: st133377@student.spbu.ru, тел.: +7(904)146-02-15

Перед считыванием электрических импульсов необходимо нанести электродный гель либо на электрод, либо на тело человека для переноса электрического тока от кожи к датчикам.

Сигнал ЭМГ имеет шумовой характер с широким спектром, и визуально анализировать его достаточно сложно. Электромиограмма содержит большое количество высокочастотных компонент. Поэтому при регистрации сигналов ЭМГ выбирают широкий частотный диапазон: верхняя полоса пропускания составляет 10–20 кГц, а нижняя 2–10 Гц.

После необходимо соединить датчики к считывающей плате и приступить к разбору сигнала. При первом считывании будет присутствовать зашумлённость (см. рис. 1). Для того чтобы избавиться от шума, применяются фильтры низких и высоких частот.

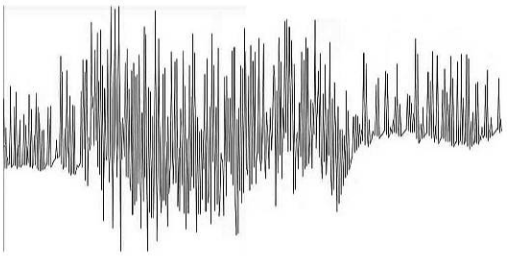


Рис. 1. Зашумленный сигнал

Для обработки входных сигналов на платформе Arduino были использованы два типа фильтров: фильтр низких частот (Low-Pass Filter) и фильтр высоких частот (High-Pass Filter). Оба фильтра реализованы с помощью рекурсивных алгоритмов, что позволяет эффективно фильтровать сигналы в реальном времени и обойтись без создания физических пассивных или активных фильтров [6, 7].

Фильтр низких частот реализуется с помощью рекурсивного цикла:

```
LP = kLow * Value + (1 - kLow) * LP,
```

где `LP` – это выходной сигнал, обработанный фильтром, `Value` – текущее значение входного сигнала, `kLow` – коэффициент фильтра, регулирующий степень фильтрации.

Данный фильтр позволяет минимизировать влияние быстропереходящих колебаний. Задача фильтра – ослабить высокочастотные колебания.

Фильтр высоких частот также реализован с использованием рекурсивного цикла:

```
HP = kHigh * (HP + Value - previousValue),
```

где `HP` – это выходной сигнал, обработанный фильтром, `Value` – текущее значение сигнала на входе, `previousValue` – значение входного сигнала на предыдущем шаге, `kHigh` – коэффициент фильтра, определяющий степень влияния изменения сигнала.

Фильтр выделяет быстродействующие компоненты и устраняет низкочастотные шумы, такие как дыхание.

Результат работы фильтров можно увидеть на рис. 2.

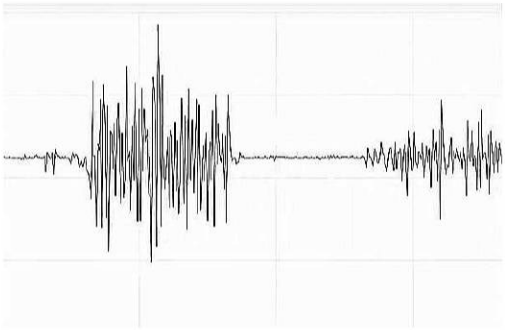


Рис. 2. Отфильтрованный сигнал

5. Классификация действий. После выделения полезного сигнала используется триггерная система, для которой строятся пороговые значения амплитуды или частоты, чтобы детектировать моменты, когда происходят определенные действия. Система реагирует на изменение сигнала по времени (см. рис. 3). Путем правильной оценки амплитуды или частоты происходит идентификация определенного события [8, 9].

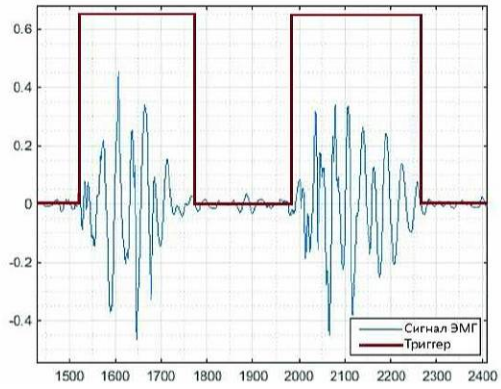


Рис. 3. Пример реализации назначения триггера на действие

6. Выводы. Возможность анализа ЭМГ сигналов предоставляет перспективную технологию для создания устройств, которые смогут обеспечить прямое и интуитивное управление устройствами.

Основным преимуществом использования ЭМГ сигнала в сочетании с платой Arduino является относительно низкая стоимость. Однако, в процессе работы необходима точная калибровка оборудования и точность снятия сигнала.

В дальнейшем существует возможность добавить считывание ЭЭГ к устройству и создать гибридную и более гибкую систему, что позволит увеличить количество функций.

Литература

1. Абрамов А. А. Аппаратно-программный комплекс для управления протезом кисти на основе сигналов электромиографии // Медицина. Социология. Философия. Прикладные исследования. 2023. № 3. С. 212–216.

2. Гаврилов А. И., Со У. Применение данных электромиографии в системах управления экзоскелетными устройствами // Инженерный журнал: наука и инновации. 2017. № 6. С. 1–9.

3. Гончарова А. Б. Постановка предварительного медицинского диагноза на основе теории нечетких множеств с использованием меры Сугено // Вестник Санкт-Петербургского университета.

Прикладная математика. Информатика. Процессы управления. 2019. Т. 15. № 4. С. 529–543.

4. Кабанов А. А. Разработка системы классификации жестов руки на основе сигналов электромиографии // Омский научный вестник. 2021. № 3. С. 79—84.

5. Легкова И. А. Обнаружение отклонений и нарушений сердечного ритма в данных ЭКГ пациентов // Процессы управления и устойчивость. 2024. Т. 11. № 1. С. 150–154.

6. Лукьянчиков А. И., Мельников А. О. и др. Алгоритмы классификации одноканального ЭМГ-сигнала для человеко-компьютерного взаимодействия // Cloud of science. 2018. № 2. С. 300–309.

7. Simpetru R. C., Cnejevici V. et al. Influence of spatio-temporal filtering on hand kinematics estimation from high-density EMG signals // Journal of Neural Engineering. 2024. Vol. 21. No 2. Art. no 026014.

8. Sadiq A., Aimal K. et al. Machine learning and signal processing based analysis of semg signals for daily action classification // IEEE Access. 2022. Vol. 10. P. 40506–40516.

9. Tepe C., Demir M C. Real-time classification of emg myo armband data using support vector machine // IRBM. 2022. Vol. 43. No 4. P. 300–308.

УДК 004.9

Топпер А. М.

Разработка автоматизированной системы обработки амбулаторных карт пациентов

Рекомендовано к публикации доцентом Гончаровой А. Б.

1. Введение. В условиях цифровизации медицины возрастает необходимость в эффективных и безопасных системах управления медицинскими данными. Медицинские данные, включая амбулаторные карты пациентов, представляют собой ценный ресурс для автоматизации врачебных задач и снижения нагрузки на медицинский персонал.

Работа с медицинскими данными является одним из важных направлений исследований [1]. Важно не только собирать и хранить полученные данные, но и исследовать их с помощью программного обеспечения [2,3]. Медицинские данные, как числовые, так и категориальные [4], представляют собой ценные параметры не только для построения интеллектуальных систем [5,6], но и для автоматизации врачебных задач.

Целью данной статьи является разработка системы, которая автоматически извлекает данные из амбулаторных карт, анализирует их и предоставляет информацию о ближайших назначениях в удобном формате. Система также обеспечивает безопасное хранение и передачу данных с использованием современных облачных технологий.

2. Структура системы. Реализованная система структурно разделена на три компонента, указанных на рис. 1: серверная и клиентская части и облачное хранилище.

Топпер Алина Михайловна – магистрант, Санкт-Петербургский государственный университет; e-mail: st089228@student.spbu.ru, тел.: +7(921)187-03-42

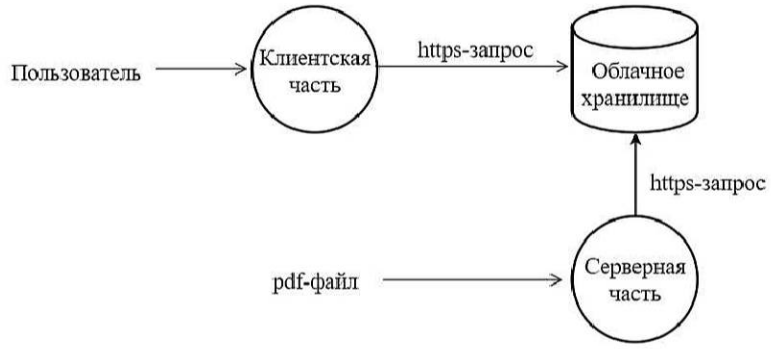


Рис. 1. Компоненты программного обеспечения

Дата осмотра 18.03.2025 ~~на приеме,~~ на дому~~, в фельдшерско-акушерском пункте, прочее~~.
Врач (специальность) Эндокринолог
Жалобы пациента Повышенная утомляемость, слабость, затруднения в контроле уровня сахара, покалывание в конечностях.
Анамнез заболевания, жизни Диагностирован сахарный диабет 2 типа в течение 5 лет, ранее проводилась коррекция питания; семейная история диабета; ранее неоднократно посещал эндокринолога
Объективные данные Уровень глюкозы – 9,8 ммоль/л, АД 130/80 мм рт. ст., имеются признаки периферической невропатии (сниженная чувствительность в нижних конечностях).
Диагноз основного заболевания: Сахарный диабет 2 типа код по МКБ-10 E11.9
Осложнения: Начало периферической невропатии
Сопутствующие заболевания Гипертония код по МКБ-10 I10
Ожирение код по МКБ-10 E66.9
Гиперлипидемия код по МКБ-10 E78.5
Внешняя причина при травмах (отравлениях) не выявлена
код по МКБ-10
Группа здоровья 3 Диспансерное наблюдение проводится

Назначения (исследования, консультации)	Лекарственные препараты, физиотерапия
Общий анализ крови – 01.04.25 в 8:00 в 1 кабинете	Продолжение приёма метформина
Консультация диетолога – 03.03.25 в 14:50 в 140 каб	Коррекция инсулиновой терапии
Корректировка терапевтического плана – 03.03.25 в 17:00 в 207 кб	Рекомендации по умеренной физической активности.
Листок нетрудоспособности, справка	Льготные рецепты
Выдан на 5 календарных дней.	Оформлены.
Информированное добровольное согласие на медицинское вмешательство, отказ от медицинского вмешательства Получено.	
Врач д-р Петрова А.В.	

Рис. 2. Пример шаблона амбулаторной карты

Преобразование pdf-файла в текст:

```
Дата осмотра
Врач (специальность)
Жалобы пациента
18.03.2025
 на приеме, на дому, в фельдшерско-акушерском пункте, прочее.
Эндокринолог
|Назначения (исследования, консультации)|Лекарственные препараты, физиотерапия|
|Общий анализ крови – 01.04.25 в 8:00 в 1 кабинете|Продолжение приёма метформина|
|Консультация диетолога – 03.03.25 в 14:50 в 140 каб|Коррекция инсулиновой терапии|
|Корректировка терапевтического плана – 03.03.25 в 17:00 в 207 кб|Рекомендации по умеренной физической активности.|
|Листок нетрудоспособности, справка|Льготные рецепты|
|Выдан на 5 календарных дней.|Оформлены.|
|Информированное добровольное согласие на медицинское вмешательство, отказ от медицинского вмешательства Получено.|None|
|Врач д-р Петрова А.В.|None|
```

Выделение сущностей из текста:

```
Найденные назначения:
- Общий анализ крови – 01.04.25 в 8:00 в 1 кабинете
- Консультация диетолога – 03.03.25 в 14:50 в 140 каб
- Корректировка терапевтического плана – 03.03.25 в 17:00 в 207 кб
```

Создание JSON-объекта:


```
{
    "patient_id": "3b27873c-a78b-4272-aec4-fab901483b16",
    "appointments": [
        {
            "description": "Общий анализ крови",
            "date": "01.04.25",
            "time": "8:00",
            "cabinet": "1"
        },
        {
            "description": "Консультация диетолога",
            "date": "03.03.25",
            "time": "14:50",
            "cabinet": "140"
        },
        {
            "description": "Корректировка терапевтического плана",
            "date": "03.03.25",
            "time": "17:00",
            "cabinet": "207"
        }
    ]
}
```


Рис. 3. Этапы преобразования pdf-файла

2.1. Серверная часть. Серверная часть программы принимает на вход амбулаторную карту пациента в формате pdf (см. рис. 2). С использованием библиотек для обработки pdf-файлов извлекается текстовая информация. Далее автоматически анализируется текст с помощью регулярных выражений, выделяются ключевые данные: имя пациента, описание назначения, дата и время. Эти данные структурируются в JSON-файл, который автоматически загружается в выбранное облачное хранилище. Примеры результатов каждого этапа обработки показаны на рис. 3.

В системе организована работа с программно сгенерированными

pdf и отсканированными документами, обработанными с помощью OCR для извлечения текста. Для работы с pdf применяются библиотеки PyPDF2 и pdfplumber, которые позволяют извлекать текст и обрабатывать таблицы. Эти pdf содержат структурированные элементы, легко редактируемые и доступные для поиска. Для обработки отсканированных документов используется библиотека pytesseract, которая преобразует изображения в текст с помощью технологии OCR, позволяя редактировать текст, сохраняя оригинальное изображение.

2.2. Облачное хранилище. Для хранения данных выбрано хранилище, которое обеспечивает высокий уровень безопасности и соответствует требованиям законодательства. Данные передаются по защищенному протоколу HTTPS, а доступ к ним ограничен с использованием OAuth 2.0. Это позволяет минимизировать риск утечки информации.

2.3. Клиентская часть. Клиентская часть реализована как веб-сервис на основе Flask. Веб-приложение получает JSON-файл с облачного хранилища, анализирует данные и находит ближайшее по времени назначение. Для озвучивания информации используется библиотека pyttsx3, которая обеспечивает синтез речи на устройстве пользователя.

3. Защита данных. Для хранения медицинских данных было выбрано облачное хранилище, обеспечивающее высокий уровень конфиденциальности и защиты информации. Платформа использует безопасные протоколы передачи и многофакторную аутентификацию, что снижает риск несанкционированного доступа.

Помимо самой безопасности облачного хранилища, для защиты данных были использованы следующие подходы:

1. OAuth 2.0: протокол авторизации для безопасного доступа к данным без раскрытия паролей.
2. Разделение доступа: доступ к JSON с назначениями ограничен только администраторами.
3. Отсутствие хранения данных на клиентской части: все данные обрабатываются в памяти, без записи на устройства клиента.

4. Заключение. Разработана система автоматизации обработки амбулаторных карт пациентов и выделения сущностей. Она позволяет снизить нагрузку на медицинский персонал, предоставляя информацию о ближайших назначениях в удобном формате. Особое внимание в системе уделено вопросам безопасности данных, что делает систему соответствующей требованиям законодательства.

Литература

1. Камгуя Ф. Х., Козубова К. В. и др. Создание набора видеозаписей ультразвуковых исследований для диагностирования очаговых образований в печени с целью обучения сервисов искусственного интеллекта // Современные методы прикладной математики, теории управления и компьютерных технологий (ПМТУКТ-2022). 2022. С. 29–30.

2. Гончарова А. Б., Виль М. Ю. Программное обеспечение для анализа медицинских данных // Современные методы прикладной математики, теории управления и компьютерных технологий (ПМТУКТ-2022). 2022. С. 13–15.

3. Kurniawati A. C. D. Telemedicine during COVID-19 pandemic: a review of the literature // Процессы управления и устойчивость. 2022. Т. 9. № 1. С. 156–164.

4. Топпер А. М. Модель прогнозирования по числовым и категориальным данным // Современные методы прикладной математики, теории управления и компьютерных технологий (ПМТУКТ-2024). 2024. С. 92–93.

5. Бусько Е. А., Гончарова А. Б. Программная реализация системы принятия диагностических решений на основе мультипараметрических ультразвуковых показателей образований молочной железы // Сибирский журнал клинической и экспериментальной медицины. 2020. Т. 35. № 4. С. 137–142.

6. Гончарова А. Б. Постановка предварительного медицинского диагноза на основе теории нечетких множеств с использованием меры Сугено // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2019. Т. 15. № 4. С. 529–543.

УДК 519.86

Шайхисламова А. Р.

Математическое моделирование динамики смертности в Республике Молдова: вероятностно-статистический анализ

Рекомендовано к публикации доцентом Гасратовой Н. А.

1. Введение. Смертность и заболеваемость – актуальные темы для научного анализа, так как оказывают значительное влияние на организацию системы здравоохранения, демографические изменения, социально-экономические факторы. Статистические данные позволяют оценить будущую демографическую ситуацию; выявить факторы риска, влияющие на продолжительность жизни; планировать медицинские расходы и социальные программы; оценивать ожидаемую нагрузку на пенсионные фонды [1–3].

В качестве объекта исследования были рассмотрены заболевания по трем системам в Республике Молдова: кровеносная, пищеварительная и нервная, которые по данным Всемирной Организации Здравоохранения (ВОЗ) занимают ведущие позиции среди причин смертности в странах Европы [4].

2. Постановка задачи и методы исследования. В исследовании были использованы данные ВОЗ и Европейский портал информации здравоохранения [4, 5].

Распределение заболеваний по системам было осуществлено по регламенту ВОЗ: а) кровеносная система: ишемическая болезнь сердца, цереброваскулярные заболевания (заболевания сосудов головного мозга), гипертония, ревматическая болезнь сердца, воспалительные и неуточненные заболевания системы кровообращения, а также другие болезни этой группы; б) пищеварительная система: цирроз печени, пептическая язвенная болезнь, панкреатит, заболевания желчного пузыря и желчевыводящих путей, кишечная непроходимость, аппендицит, гастрит и дуоценит, воспалительные и другие заболевания органов пищеварения; в) нервная система: алкоголизм,

Шайхисламова Алина Ринатовна – магистрант, Санкт-Петербургский государственный университет; e-mail: st091115@student.spbu.ru. тел.: +7(931)260-23-34

болезнь Альцгеймера и другие виды деменции, эпилепсия, расстройства, связанные с употреблением наркотиков, болезнь Паркинсона, рассеянный склероз, шизофрения, униполярные депрессивные расстройства, биполярное аффективное расстройство [4].

Данные по смертности представлены в соответствии с возрастной группой (таблица 1) и полом (все, женщины, мужчины).

Таблица 1. Номер группы в соответствии с возрастом

Возрастная группа	**Номер**
[0]	0
[1–4]	1
[5–9]	2
[10–14]	3
[15–19]	4
[20–24]	5
[25–29]	6
[30–34]	7
[35–39]	8
[40–44]	9
[45–49]	10
[50–54]	11
[55–59]	12
[60–64]	13
[65–69]	14
[70–74]	15
[75–79]	16
[80–84]	17
[85+]	18

Анализ данных проводился с помощью программного пакета Excel и языка программирования Python.

3. Основные демографические тренды Республики Молдова. Демографическая и социально-экономическая ситуация Республики Молдова характеризуется рядом проблем, которые часто встречаются и в других странах: снижение рождаемости, увеличение нетрудоспособного населения, гендерный дисбаланс. Поэтому данный пример может быть показателен при изучении смертности с учетом пола, возрастной группы и года. Кроме того, Молдову можно отнести к странам, экономика которых испытывает значительные сложности [6].

4. Математическое моделирование смертности. Для анализа возрастной динамики смертности и сравнения различных систем применялись математические модели.

4.1. Моделирование динамики смертности при помощи логистической функции. Смертность по возрастным группам можно описать с помощью логистической функции

$$f(t) = \frac{L}{1 + e^{-k(t-t_{\max})}},$$

где $f(t)$ – значение смертности для возрастной группы t, L – максимальное значение смертности, k – коэффициент роста, определяющий скорость увеличения смертности с возрастом, $t_{\max}$ – возрастная группа с максимальной скоростью роста смертности.

Параметры L, k и $t_{\max}$ определялись с помощью метода наименьших квадратов (МНК), минимизирующего расхождения между расчетными и эмпирическими данными.

4.2. Применение основ теории надежности. Интенсивность отказа системы. Подходы теории надежности к моделированию биологических процессов широко используются в геронтологии и демографии для описания процессов старения и смертности [7]. Человеческий организм можно рассматривать как сложную систему, что позволяет применять методы теории надежности для анализа вероятности отказа, где отказом считается наступление смерти [8]. Экспоненциальная модель, применяемая в данной работе, описывает накопленную вероятность отказа системы (смерти) к определённому возрасту. Эта модель представлена следующим уравнением:

$$R(t) = 1 - e^{-\lambda t},$$

где $R(t)$ – функция накопленной вероятности отказа системы, определяющая вероятность её выхода из строя к возрасту t, λ – интенсивность отказов (скорость выхода из строя), t – средний возраст в каждой возрастной группе. Для оценки параметра λ, характеризующего интенсивность отказов, применялся МНК. Эмпирическая функция отказа рассчитывалась как отношение вида:

$$R(t) = \frac{\text{Кумулятивное количество умерших к } t}{\text{Общее количество смертей за рассматриваемый год}}$$

Все расчёты проводились отдельно для трёх исследуемых систем, для каждого среднего возраста t с 2002 по 2018 год. Далее по полу-

ченным данным вычислялся средневзвешенный параметр интенсивности отказа λ.

5. Результаты. В ходе исследования были получены параметры L, k, $t_{\max}$ по МНК, которые представлены в таблице 2.

Таблица 2. Параметры для функции логистической аппроксимации

Система	**Год**	**Пол**	L	k	$t_{\max}$
Кровеносная	2002	Все	22041,37	1,25	16,56
Кровеносная	2002	Женский	20648,04	1,32	16,57
Кровеносная	2002	Мужской	25817,82	1,20	16,62
Кровеносная	2007	Все	17127,31	1,08	16,33
Кровеносная	2007	Женский	16868,78	1,19	16,42
Кровеносная	2007	Мужской	17726,49	0,96	16,19
Кровеносная	2012	Все	15236,25	1,13	16,28
Кровеносная	2012	Женский	15405,27	1,26	16,45
Кровеносная	2012	Мужской	14829,98	1,01	15,94
Кровеносная	2017	Все	13735,08	1,28	16,34
Кровеносная	2017	Женский	14381,98	1,41	16,53
Кровеносная	2017	Мужской	12349,47	1,15	15,88
Пищеварительная	2002	Все	450,03	0,82	10,89
Пищеварительная	2002	Женский	420,97	0,91	11,23
Пищеварительная	2002	Мужской	497,26	0,78	10,59
Пищеварительная	2007	Все	455,52	0,76	10,85
Пищеварительная	2007	Женский	440,55	0,84	11,29
Пищеварительная	2007	Мужской	475,78	0,75	10,37
Пищеварительная	2012	Все	366,12	0,80	10,81
Пищеварительная	2012	Женский	352,05	0,82	11,49
Пищеварительная	2012	Мужской	389,65	0,84	10,18
Пищеварительная	2017	Все	317,74	0,70	11,25
Пищеварительная	2017	Женский	339,60	0,58	12,57
Пищеварительная	2017	Мужской	321,52	0,93	10,36
Нервная	2002	Все	28,02	1,28	6,90
Нервная	2002	Женский	16,17	2,03	7,65
Нервная	2002	Мужской	44,15	1,23	6,86
Нервная	2007	Все	38,31	1,70	7,76
Нервная	2007	Женский	22,21	1,22	8,41
Нервная	2007	Мужской	59,23	2,07	7,82
Нервная	2012	Все	28,26	1,35	7,60
Нервная	2012	Женский	14,14	1,04	7,89
Нервная	2012	Мужской	47,78	1,38	7,74
Нервная	2017	Все	21,84	0,87	7,45
Нервная	2017	Женский	11,69	0,30	5,71
Нервная	2017	Мужской	36,06	1,01	7,82

Анализ параметра L показал устойчивую тенденцию к снижению по всем исследуемым системам заболеваний в период с 2002 по 2017 год.

Параметр $t_{\max}$ указывает на возрастную группу, начиная с которой отмечается наиболее резкий рост смертности. Таким образом, моделирование показало, что в среднем по всем годам, для кровеносной системы к группе риска относится 16 группа (75–79 лет); для пищеварительной – 11 группа (50–54 лет); для нервной – 7–8 группы (30–39 лет). Полученные результаты согласуются с данными ВОЗ [4].

Параметр k характеризует скорость роста смертности с возрастом. Как видно из таблицы 2, его значения варьируются: при высоких k смертность возрастает резче, при низких – медленнее. Например, в 2002 году для нервной системы $k = 1{,}28$, для кровеносной $k = 0{,}71$, что отражает различия в динамике смертности между системами.

На следующем этапе анализа были найдены средневзвешенные коэффициенты интенсивности отказа, представленные в таблице 3.

Таблица 3. Интенсивность отказа λ для каждой системы в Республике Молдова по данным смертности с 2002 по 2018 года

Система	**Параметр λ**
Кровеносная	0,00390
Пищеварительная	0,01107
Нервная	0,01850

Анализ интенсивности отказов показал, что максимальное значение λ наблюдается у нервной системы, что свидетельствует о высокой скорости нарастания смертности. Пищеварительная система имеет средний уровень интенсивности отказов, тогда как минимальные значения у кровеносной системы указывают на более плавное увеличение смертности с возрастом, что соответствует накопительному характеру сердечно-сосудистых заболеваний.

Литература

1. Смирнова Т. М., Крутько В. Н. Историческая динамика смертности и ее учет в целях стратегического планирования медицинской и социальной помощи пожилым // Клиническая геронтология. 2018. Т. 24. № 9–10. С. 63–65.

2. Синдяшкина Е. Н. Ожидаемая продолжительность здоровой жизни в контексте десятилетия здорового старения ООН // Анализ и прогноз. Журнал ИМЭМО РАН. 2022. № 1. С. 40–53.

3. Цветков А. Л. Эмпирические модели, описывающие долю населения старше 65 в странах Европы // Процессы управления и устойчивость. 2018. Т. 5. № 1. С. 523–526.

4. WHO Mortality Database [Электронный ресурс]: https://www.who.int/data/data-collection-tools/who-mortality-database (дата обращения: 20.10.2024).

5. European Health for All database (HFA-DB) [Электронный ресурс]: https://gateway.euro.who.int/en/country-profiles/republic-of-moldova/ (дата обращения: 11.01.2025).

6. Экономические показатели. Trading Economics [Электронный ресурс]: https://tradingeconomics.com/matrix (дата обращения: 10.03.2025).

7. Крутько В. Н., Славин М. Б., Смирнова Т. М. Математические основания геронтологии: Общая теория здоровья; теория надежности в живых и неживых системах; современные методы анализа биологического возраста, старения и продолжительности жизни. М.: Едиториал УРСС, 2002. 384 с.

8. Чистяков В. А., Денисенко Ю. В., Празднова Е. В., Емельянцев С. А. Старение и надежность: как вариации во времени интенсивности отказов влияют на форму кривых дожития стареющих организмов // Известия высших учебных заведений. Северо-Кавказский регион. Естественные науки. 2024. № 2. С. 144–155.

УДК 004.942

Шарашева Н. И.

Прогнозирование распространенности онкологических заболеваний

Рекомендовано к публикации доцентом Лежниной Е. А.

1. Введение. В большинстве стран с высоким и средним уровнем экономического развития онкологические заболевания занимают первые позиции среди причин смерти людей. В России они занимают 2 место месте после болезней системы кровообращения, при этом заболеваемость раком возрастает в среднем на 2–3% в год.

Прогнозирование заболеваемости онкологией играет ключевую роль в современном здравоохранении, позволяя не только оценивать будущую динамику болезни, но и разрабатывать эффективные стратегии профилактики, диагностики и лечения. Точные прогнозы необходимы для оптимального планирования медицинской инфраструктуры и распределения ресурсов, а также помогают оценить экономическое бремя рака для общества, включая затраты на лечение и потери трудоспособности.

2. Обзор методов. Данные о заболеваемости и смертности представляют собой временной ряд, основным источником которого являются статистические ежегодники [1]. Историческая глубина доступных данных ограничена 30–40 годами, что существенно снижает эффективность применения сложных моделей, таких как глубокие нейронные сети или гауссовские процессы. В условиях ограниченного объема выборки такие модели подвержены риску переобучения, что приводит к снижению их обобщающей способности. В подобных сценариях они не всегда обеспечивают значимое преимущество в точности прогноза по сравнению с более простыми статистическими методами, которые оказываются более устойчивыми и надёжными при работе с малым числом наблюдений.

Одним из самых распространенных методов для прогнозирования одномерных временных рядов являются модели скользящего среднего ARIMA(p, d, q) [2], [3]. Эта модель сочетает три компонента:

Шарашева Надежда Ильдаровна – студент, Санкт-Петербургский государственный университет; e-mail: st097214@student.spbu.ru, тел.: +7(917)883-07-83

авторегрессию (AR), интегрированный компонент (I) и скользящее среднее (MA).

Авторегрессия учитывает зависимость текущего значения ряда от предыдущих наблюдений. Интегрированный компонент устраняет тренд путем дифференцирования, что делает ряд стационарным. Скользящее среднее описывает зависимость текущего значения от ошибок прошлых предсказаний, и учитывает краткосрочные отклонения от общей тенденции.

Модель зависит от 3 гиперпараметров: p – порядок авторегрессии, d – порядок дифференцирования и q – порядок скользящего среднего. ARIMA применяется для стационарных временных рядов, но может быть адаптирована для работы с нестационарными данными за счёт включения дифференцирования ($d > 0$), что подробно описано в работе [4]. Также эффективно показывают себя модели экспоненциального сглаживания Холта – Уинтерса [5].

В работе [6] рассмотрено прогнозирование распространения пандемии COVID-19 с помощью интегральной модели притока – оттока. Также данная модель была использована для прогнозирования численности населения Земли в работе [7]. Предлагается использовать аналогичный подход для построения прогнозов распространенности неинфекционных заболеваний, взяв онкологию в качестве примера.

3. Модель для прогноза численности населения, подверженного онкологическим заболеваниям. Интегральная модель притока и оттока описывает динамику численности пациентов с онкологическими заболеваниями в популяции, учитывая приток новых случаев (диагностированные пациенты) и отток (умершие и выздоровевшие пациенты). Модель основана на принципе динамического баланса, согласно которому общее количество пациентов в любой момент времени определяется разницей между числом вновь выявленных случаев и числом пациентов, выбывших из группы больных. В случае онкологических заболеваний будем рассматривать только число умерших пациентов в качестве оттока, поскольку при выходе в ремиссию сохраняется риск повторного заболевания, и пациенту необходимо продолжать наблюдения у врача, т. е. пациент не выходит из группы больных.

Пусть $X(t)$ – число людей, подвергнувшихся онкологическим заболеваниям в момент времени t. В каждый момент времени поступает $x_{\text{inf}}(t)$ новых пациентов и выбывает $x_{\text{of}}(t)$ пациентов. Тогда со-

стояние системы можно описать дискретным уравнением

$$X(t) = X(t-1) + x_{\text{inf}}(t) - x_{\text{of}}(t).$$

Приток и отток пациентов являются стохастическими процессами, поэтому эволюция системы $X(t)$ становится случайным процессом, зависящим от временных рядов $x_{\text{inf}}(t)$ и $x_{\text{of}}(t)$.

Введем интегральные значения притока и оттока как суммы членов соответствующих временных рядов

$$X_{\text{inf}}(t) = \sum_{\tau=t_0}^{t} x_{\text{inf}}(\tau), \quad X_{\text{of}}(t) = \sum_{\tau=t_0}^{t} x_{\text{of}}(\tau).$$

Тогда основное уравнение модели примет вид

$$X(t) = X(t_0) + X_{\text{inf}}(t) - X_{\text{of}}(t).$$

Для анализа динамики притока и оттока вводятся процентные приросты

$$r_{\text{inf}}(t) = \frac{X_{\text{inf}}(t) - X_{\text{inf}}(t-1)}{X_{\text{inf}}(t-1)} \cdot 100\%,$$
$$r_{\text{of}}(t) = \frac{X_{\text{of}}(t) - X_{\text{of}}(t-1)}{X_{\text{of}}(t-1)} \cdot 100\%.$$

Рассмотрим модель, описывающую динамику онкологических заболеваний при $t > t_0$, представленную системой дискретных уравнений:

$$C(t) = \sum_{t=t_0+1}^{t} x_{\text{inf}}(t),$$
$$R(t) = \sum_{t=t_0+1}^{t} x_{\text{of}}(t).$$
$$I(t) = C(t) - R(t),$$

где $I(t)$ – число людей, подверженных заболеванию, $C(t)$ и $R(t)$ – это кумулятивные суммы новых случаев и смертей соответственно, начиная с момента времени t_0+1 до момента времени t. Для получение прогнозов используются рекуррентные формулы

$$C(t) = \left(1 + \frac{r_{\text{inf}}(t)}{100}\right) C(t-1), \tag{1}$$

$$R(t) = \left(1 + \frac{r_{\text{of}}(t)}{100}\right) R(t-1), \tag{2}$$

$$I(t_0) = C(t_0) = R(t_0) = 0. \tag{3}$$

Параметрами модели являются процентный прирост количества выявленных случаев заболевания $r_{\text{inf}}(t)$ и процентный прирост количества смертей $r_{\text{of}}(t)$, изменяющиеся во времени.

Модель (1)–(3) является интегральной моделью притока – оттока пациентов с онкологическими заболеваниями.

4. Численный эксперимент. Был выполнен прогноз численности людей, подверженных онкологическим заболеваниям, на основе данных за 1993–2023 годы.

Точность модели оценивалась по предсказаниям количества пациентов на 2017–2023 года с помощью метрики

$$\text{MAPE} = \frac{1}{n} \sum_{t=1}^{n} \left| \frac{I(t) - \widehat{I}(t)}{I(t)} \right| 100\%.$$

Оценка MAPE ретроспективных прогнозов с помощью интегральной модели притока-оттока равна 9,5%. Для сравнения, сделан прогноз с помощью модели ARIMA(7,1,6). Процесс подбора гиперпараметров модели ARIMA начинается с определения порядка дифференцирования d. Для этого используется тест Дики – Фуллера, позволяющий выявить необходимость преобразования временного ряда в стационарный вид. После приведения ряда к стационарному состоянию осуществляется выбор порядка авторегрессии p — он определяется на основе графика частичной автокорреляционной функции, где ключевым ориентиром служит резкий спад значений. Аналогичным образом порядок компоненты скользящего среднего q устанавливается по автокорреляционной функции, анализируя тот лаг, на котором происходит значительное снижение корреляции. Значение MAPE для модели ARIMA(7,1,6) равно 11,09%.

Для прогнозирования заболеваемости на период 2023–2030 годы были построены аппроксимирующие степенные функции для параметров $r_{\text{inf}}(t)$ и $r_{\text{of}}(t)$ на основе данных за 1993–2023 годы. В таблице 1 приведены полученные функции и значения коэффициента

детерминации, характеризующего качество аппроксимации и вычисленного по формуле

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \overline{y})^2},$$

где y_i – реальные значения, $\widehat{y}_i$ – предсказанные моделью значения, $\overline{y}$ – среднее значение исходных данных, n – количество наблюдений.

Таблица 1. Аппроксимация параметров на период 2023–2030

Параметр	**Аппроксимирующая функция**	$\mathbf{R^2}$
$r_{\text{inf}}(t)$	$94{,}2x^{-0{,}93}$	0,9989
$r_{\text{of}}(t)$	$97{,}957x^{-0{,}874}$	0,9981

Таблица 2. Прогноз на период 2023–2030

Дата	**ИП, чел.**	**Прирост ИП, %**	**ИО, чел.**	**Прирост ИО, %**	**I(t), чел.**
2023	15346067	3,30064	9993779	4,0455	5352288
2024	15912409	3,690471	10467218	4,737338	5445191
2025	16482909	3,585252	10949927	4,611628	5532982
2026	17057510	3,486042	11441892	4,492860	5615618
2027	17636158	3,392335	11943100	4,380463	5693058
2028	18218801	3,303683	12453539	4,273927	5765262
2029	18804541	3,215031	12972527	4,167391	5832014
2030	19392442	3,126379	13499322	4,060855	5893120

5. Заключение. Результаты проведенных вычислительных экспериментов подтвердили высокую эффективность интегральной модели притока-оттока для прогнозирования распространенности онкологических заболеваний. Эта модель дала более точные прогнозы по сравнению с методами, основанными на модели ARIMA, что позволяет делать более обоснованные выводы о будущем развитии ситуации.

В таблице 2 представлен прогноз на период 2023–2030 годы, который показывает устойчивый рост численности пациентов с онкологическими заболеваниями, что подчеркивает необходимость дальнейшего развития стратегий профилактики, диагностики и лечения. Полученные результаты можно использовать для оценки демографического и экономического ущербов, путем расчета экономического бремени заболевания и количества потерянных лет человеческих жизней.

Литература

1. Заболеваемость онкологией и смертность от нее с 1997 года // МНИОИ им. П.А. Герцена; обработка «Если быть точным», 2024. [Электронный ресурс]: URL:https://tochno.st/datasets/cancer_incidence (дата обращения: 17.10.24).

2. Jian Y., Zhu D., Zhou D. ARIMA model for predicting chronic kidney disease and estimating its economic burden in China // BMC Public Health. 2022. No 22. [Электронный ресурс]: URL:https://doi.org/10.1186/s12889-022-14959-z (дата обращения: 24.10.24).

3. Оленев Р. О. Прогнозирование фондовых индексов с использованием моделей ARIMA и LSTM // Процессы управления и устойчивость. 2023. Т. 10. № 1. С. 304–308.

4. Box G. E. P., Jenkins G. M., Reinsel G. C., Greta M. L. Time Series Analysis Forecasting and Control. New Jersey: John Wiley and Sons Inc., 2015. 712 p.

5. Wang N., Zhuang W., Ran Z. et al. Prediction of acute onset of chronic cor pulmonale: comparative analysis of Holt-Winters exponential smoothing and ARIMA model // BMC Med. Res. Methodol. 2024. No 204. [Электронный ресурс]: URL:https://doi.org/10.1186/s12874-024-02325-z (дата обращения: 24.10.24).

6. Балыкина Ю. Е., Захаров В. В. Интегральная модель притока и оттока и ее приложения // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2024. Т. 20. № 2. С. 121–135.

7. Аникушин В. А., Редько А. Ю. Применение модели ARIMA для прогнозирования численности населения Земли // Процессы управления и устойчивость. 2024. Т. 11. № 1. С. 229–306.

UDC 004.932

Kamguia F. H.

AI-explainable medical image comparison

Recommended for publication by associate professor Goncharova A. B.

1. Introduction. Medical imaging refers to various techniques used to reveal internal structures or organs beneath the skin. It can also be used to study the structure and function of bones in the human body. Facilitates early detection, treatment planning, and monitoring. This field has transformed healthcare by providing critical information about anatomical, physiological, and pathological processes. Medical images are made up of various modalities that are related to the equipment and techniques used to capture them. X-rays are a technique used to generate tissues and structures in the human body, particularly when investigating bone fractures. Computed tomography (CT) produces detailed cross-sectional images that are useful for detecting tumors, vascular diseases, and injuries. Magnetic Resonance Imaging (MRI) uses magnetic fields and radio waves to generate high-resolution images of soft tissues. Positron Emission Tomography (PET) provides functional imaging by assessing metabolic processes. Ultrasound is a technique used in obstetrics and cardiology to image soft tissues in real time using high-frequency sound waves. Each of these methods produces images with different specificity and characteristics,which are called modalities. The processing of medical images remains crucial to today, as it plays a vital role in current medical diagnostic methods. [1]

Artificial intelligence techniques are increasingly being used in solving a wide range of medical problems. This is also true in the diagnosis of various cancers and other diseases. However, behind the problem of identifying neoplasms or classifying medical images using artificial intelligence methods is the issue of explainable artificial intelligence. Artificial methods are commonly referred to as black boxes because it is extremely difficult to explain in detail what the model does at each step. This makes it difficult for medical professionals to fully trust or understand the results of this technique. As a result, their acceptability and integration

Kamguia Feukwi Herve – PhD student, Saint Petersburg State University; e-mail: st093241@student.spbu.ru, phone: +7(812)428-47-26

The author acknowledges Saint Petersburg State University for a research project 121061000159-6

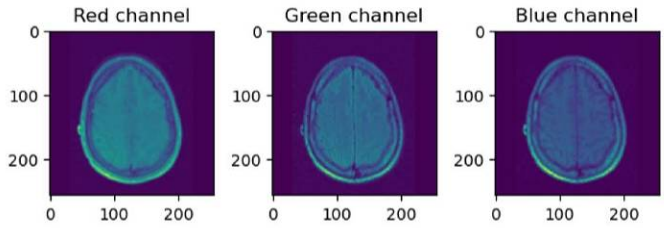
into medical systems or procedures is severely limited [2]. Furthermore, the interpretation of the results is not always well-founded or clear. In this study, our objective was to create an algorithm that would allow us to compare sequential images and better identify neoplasms in brain images of MRI fluid attenuated inversion recovery (FLAIR) for an explainable AI approach.

2. Material and methods. Here we describe the dataset we used and explain the methods we apply.

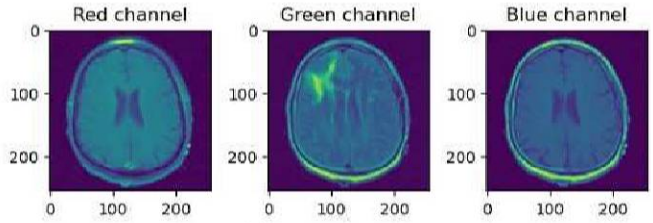
2.1. Material. The Kaggle platform is where our data set was collected [3]. Brain MRI images and manual FLAIR abnormality segmentation masks are included in this data set. The Cancer Imaging Archive (TCIA) provided the images used in this dataset. Fluid-attenuated inversion recovery (FLAIR) sequence was used to enhance the visibility of abnormalities. The data set consists of 7858 image files in total. Each picture is an RGB image created by combining an MRI slice of three modalities. Both tumor and healthy magnetic resonance (MRI) slices are included in the dataset. Patients had a variety of tumor types, such as oligodendroglioma, astrocytoma, and oligoastrocytoma.

2.2. Methods. Usually when imaging the brain, it is decided which plane to use based on medical needs. There are 3 different planes that might be considered; sagittal plane, coronal or axel plane [4]. The sagittal plane is a vertical cut that divides the brain into left and right halves. It is useful for detecting brain asymmetries, tumors, or lesions that affect the two hemispheres differently. The coronal plane is a vertical cut that divides the brain into front (anterior) and back (posterior) sections. The axial plane is a horizontal cut dividing the brain into upper (superior) and lower (inferior) parts. For our study, we chose images obtained from the sagittal plane. The human anatomy is symmetrical. This symmetrical feature is critical in medical images analysis. Its absence can indicate the presence of a pathology in a medical image. Quantification of this symmetry can then serve as an important feature in medical image processing [5]. Our method determines the symmetry axis based on the image dimension and then divide the image into halves.

An MRI slice without cancer is shown in Fig. 1a and that with cancer is shown in Fig. 1b. Every image is displayed in the Red, Green, and Blue color channels. In one channel, certain features are more noticeable than in another. For instance, in the image below, the tumor is more noticeable in the green channel. Our images were $256 \times 256 \times 3$ in size,



(a) Without cancer



(b) With Cancer

Figure 1. MRI slices samples

with the third element representing the red, green, and blue color maps. To determine which current algorithms, appear to produce the most accurate results, we tested a variety of them. The cosine-based method in our approach uses the VGG16 backbone neural network to extract a feature vector and calculates the cosine similarity between one of the halves and its flipped counterpart. Each feature vector was of dimension 25088×1. Before applying the MSE method, the image matrix data was normalized to float values. Furthermore, the SSIM method was adapted to match that of Wang et al. [6]. Here is the developed algorithm:

- start;
- separate the image into two halves;
- flip one of the images on the vertical axis;
- extract the R (red), G(green), B(blue) components;
- calculate the similarity index of each component between image;
- take the mean of the values to have the final results;
- end.

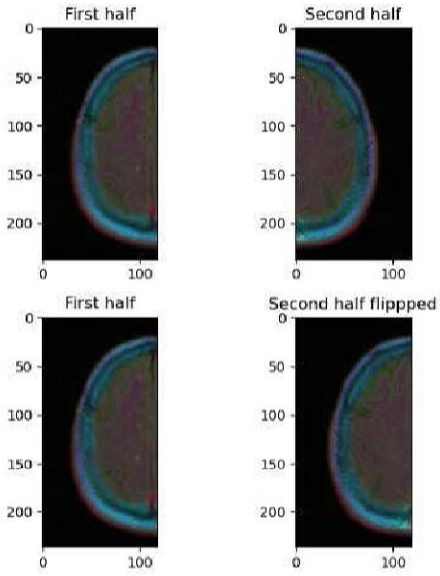


Figure 2. MRI slice without cancer

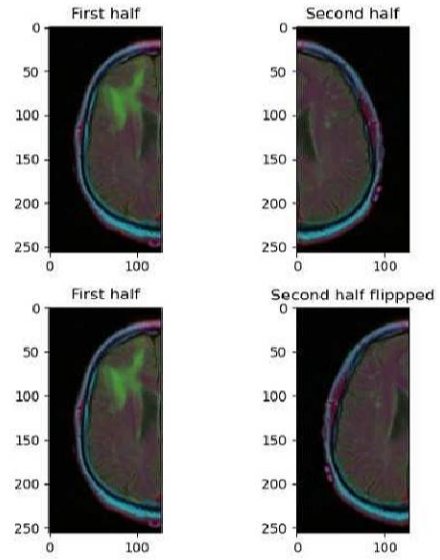


Figure 3. MRI slice with cancer

Our images are divided into two halves for comparison. To avoid biases, it is crucial that the images are divided into equal parts. The three combined channels are shown in Fig. 2, 3: flipped image halves.

3. Results. As shown in the tables, the similarity index should range from 0 to 1, with the exception of the mean square error method, which has no range. Except for MSE values, the closer the value is to 0, the more dissimilar the images are, while the closer the value is to 1, the more similar. MSE index values closer to zero indicate more similar images. SSIM and feature-based methods produce the most significant results; showing about 10% variation. The BRIEF and MSE methods exhibited little variation. It is clear from this research that when dealing with medical images, it is preferable to use SSIM or feature-based methods.

Table 1. Image without cancer

Methods	Red	Green	Blue	Mean	Range
BRIEF	0.7496	0.7566	0.792	0.7661	[0, 1]
SSIM	0.6947	0.675	0.6642	0.67797	[0, 1]
Cosine	0.7787	0.7529	0.8143	0.782	[0, 1]
MSE	0.0032	0.0026	0.0028	0.0029	–

Table 2. Image with cancer

Methods	Red	Green	Blue	Mean	Range
BRIEF	0.7185	0.7273	0.7584	0.7347	[0, 1]
SSIM	0.6013	0.5277	0.5364	0.5551	[0, 1]
Cosine	0.6916	0.6873	0.7	0.693	[0, 1]
MSE	0.004	0.011	0.01	0.0083	–

4. Discussion. Computing similarity separately for R, G, and B channels enables a more detailed and accurate comparison, particularly when color differences are significant. FlAIR MRI slices provide more visual details than grayscale MRI slices. The green channel provided more relevant visible information about the presence or absence of tumors in the MRI slices. Tumor presence was almost imperceptible to the naked eye in the red and blue channels.This image dissimilarity is visible in the tables similarity index, where the values for the green channel for the tumor image are relatively lower than those for the other channels. MSE's unsatisfactory results could be explained by the fact that it only measures absolute pixel differences and ignores edges, textures, and high-level structural features. The differences in average values indicate that the SSIM and Cosine Similarity methods will perform better in medical image processing tasks such as segmentation, classification, and tumor

identification. It is important to note that the halves used for comparison are from the same image, with nearly identical structures and pixel intensity distributions. Thus, the values indicate a strong similarity between the images. Though, we used a deep neural network to retrieve image's feature vectors, traditional computer vision methods such as color histogram features and Canny edge detection techniques can also be used.

5. Conclusion. Machine learning models can use color channel differences to improve classification and pattern recognition. Individual channel analysis detects color-based changes that grayscale may ignore. Our approach assessed similarities in each color channel compared to existing methods that used single channels. SSIM and feature-based techniques yielded more significant results than the other methods. This could be explained by the fact that the differences between cancerous and noncancerous slices are primarily texture-related. FLAIR images are primarily color and texture-based. Various existing deep learning neural networks employ the SSIM technique to solve various tasks such as image reconstruction and denoising. Others apply the cosine similarity technique to metric learning and face recognition tasks. The capacity to replicate similar results and generate saliency maps of important processes in AI-driven solutions will contribute to AI's acceptance and adoption in the healthcare domain. Setting a threshold for similarity index can be utilized as a quantitative parameter in AI image processing, increasing results and explainability [7]. Future investigations will leverage the proposed symmetry method in [5], along with our color channel approach, to test on a larger dataset.

References

1. Kotina E. D., Leonova E. B., Ploskikh V. A. Radionuclide images processing with the use of discrete systems // Vestnik of Saint Petersburg University. Applied Mathematics. Computer Science. Control Processes. 2019. Vol. 15. No 4. P. 544–554. (In Russian)

2. Mathew J., Chitra R., Stephen C., Koshy R. ntegration of Explainable Artificial Intelligence (XAI) in the Development of Disease Prediction and Medicine Recommendation System // IEEE Int. Conf. on Interdisciplinary Approaches in Tech. and Management for Social Innov. (IATMSI). 2024. P. 1–5.

3. Brain MRI Segmentation[Internet resource]: URL:https://www.kaggle.com/datasets/mateuszbuda/lgg-mri-segmentation (date: 16.03.25).

4. Reimer P., Parizel P. M., Stichnoth F.-A. Clinical MR Imaging. Berlin, Heidelberg: Springer, 2006. 597 p.

5. Hogeweg L. et al. Fast and effective quantification of symmetry in medical images for pathology detection: Application to chest radiography // Medical Physics, 2017. Vol. 44. No 6. P. 2242–2256.

6. Wang Z., Bovik A., Sheikh H., Simoncelli E.Image quality assessment: From error visibility to structural similarity // IEEE Trans. on Image Proc. 2004. Vol. 13. No 4. P. 600–612.

7. Nikitaev V., Pronichev A., Nagornov O., Sergeev V., Kruglova L., Otchenashenko A., Deeva O. Method for assessing the symmetry of the globule pattern in artificial intelligence systems for diagnosing skin neoplasms // Measuring equipment. 2024. Vol. 9. No 4. P. 53–60.

UDC 519.1

Wang B., Hu R., Xia Y.

Rumor spread model and impact assessment in public health mergencies

Recommended for publication by associate professor Eremin A. S.

1. Introduction. Public health emergencies (e.g., COVID-19 pandemic) are characterized by rapid rumor spread and severe social harm, including public panic and disrupted anti-epidemic order. Traditional rumor models rarely quantify direct impacts on public health, such as medical resource rushes and increased prevention costs [1]. This study constructs a "rumor-public health cost" linkage model to provide decision-making support for emergency management, using Livejournal network data to simulate public health scenarios.

2. Model principle.

2.1. Crowd classification (SITR Model). The population is divided into four categories [2]:

- Ignorant individuals (I): people unaware of rumors,
- Rumor spreaders (S): people spreading rumors,
- Truth spreaders (T): people disseminating factual information,
- Rumor immune (R): people aware of rumors but non-spreading.

Assuming an open network $G = (V, E)$ with N nodes, where c is the inflow rate of ignorant individuals and g is the outflow rate of all populations [3].

2.2. Propagation rules. Key interaction rules are:

1. When I contacts S: Converts to S with probability λ (rumor spread rate), T with ϵ (truth conversion rate), or R with ρ (loss of interest rate).

Wang Bowei – undergraduate student, Yan'an University,
e-mail: 1044732340@qq.com, phone: +8618089179162
Hu Ruiming – undergraduate student, Yan'an University,
e-mail: 2498016492@qq.com, phone: +8618091027741
Xia Yu – undergraduate student, Yan'an University,
e-mail: 3095284429@qq.com, phone: +8618588473760

2. When I contacts T: Converts to T with probability γ_T (truth spread rate).
3. When S contacts T: Converts to R with ω (truth suppression rate) or T with γ_R (truth refutation rate).

2.3. Public health cost function. The quadratic linear weighted model is used to quantify costs:

$$C(t) = \beta \left[aS(t) + b(1 - T(t)) + c \int_0^t S(\tau)\, d\tau \right],$$

where β is the inherent risk coefficient of public health events, a is the immediate impact weight of rumor spreaders, b is the cumulative risk weight of insufficient truth dissemination, and c is the long-term impact weight of continuous rumor spread.

2.4. Mean-field equations. Based on rumor dynamics [1], the following equations are derived:

$$\begin{cases} \dfrac{dI(t)}{dt} = c - \bar{k}I(t)S(t) - \varepsilon\bar{k}I(t)T(t) - gI(t), \\ \dfrac{dS(t)}{dt} = \lambda_S\bar{k}I(t)S(t) - (\gamma_T + \gamma_R)\bar{k}S(t)T(t) - \alpha\bar{k}S(t)(S(t) + R(t)) \\ \qquad\qquad - \delta S(t) - gS(t), \\ \dfrac{dT(t)}{dt} = \lambda_T\bar{k}I(t)S(t) + \varepsilon\bar{k}I(t)T(t) + \gamma_T\bar{k}S(t)T(t) - gT(t), \\ \dfrac{dR(t)}{dt} = \lambda_R\bar{k}I(t)S(t) + \gamma_R\bar{k}S(t)T(t) + \alpha\bar{k}S(t)(S(t) + R(t)) \\ \qquad\qquad + \delta S(t) - gR(t), \\ \dfrac{dC(t)}{dt} = \beta[aS(t) + b(1 - T(t)) + cS(t)]. \end{cases} \tag{1}$$

The definitions of coefficients in Equation (1) can be seen in Figure 1.

2.5. Propagation rules. As shown in Figure 1, When an ignorant person comes into contact with a rumor spreader, he cannot remain ignorant. He will believe the rumor with a certain probability (rumor spread rate λ_S) and become a rumor spreader; or realize that the information is false with a certain probability (truth conversion rate λ_T) and become a truth spreader; or lose interest in the rumor with a certain probability(rumor immune rate λ_R) and become a rumor immune person.

When an ignorant person comes into contact with a truth spreader, he will believe and spread the truth with a certain probability (truth spread rate ε) and become a truth spreader.

When rumor spreaders encounter truth spreaders, they will lose interest in rumors with a certain probability (truth suppression rate γ_R) and transform into rumor immune individuals, or be convinced with a certain probability (truth refutation rate γ_T) and transform into truth spreaders.

When a rumor spreader encounters other rumor spreaders or rumor immune individuals, they will lose interest in the rumor with a probability of a (rumor suppression rate α), thereby converting the rumor immune individuals into positive ones.

Due to human forgetfulness, rumor spreaders tend to forget rumors with a certain probability(forgetting rate δ) and become rumor immune individuals.

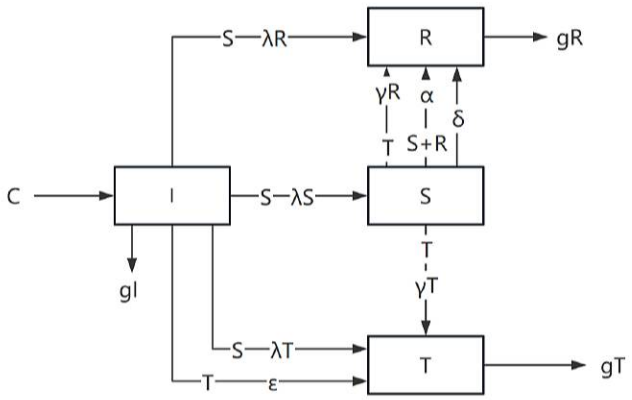


Figure 1. Diagram of the spread process

3. Numerical simulation. Simulations on Livejournal network show [4] the following.

Population dynamics: Initially dominated by I, S peaks then declines as T converts individuals, eventually leaving only R and T (Figure 2).

Cost dynamics: Costs surge initially due to panic, peak at mid-term, then decline as truth disseminates (Figure 3).

The nonlinear curve in Figure 4 reveals that small rumor scales cause

immediate costs, while large scales amplify quadratic effects.

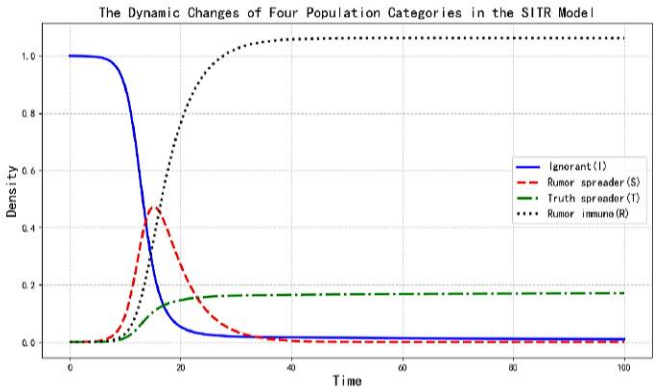


Figure 2. The dynamic changes of four population categories in the SITR model

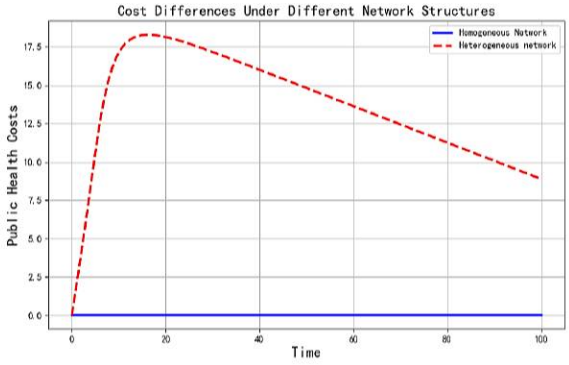


Figure 3. Cost differences under different network structures

4. Conclusions and innovations.

4.1. Main conclusions. Rumor spread exhibits nonlinear positive correlation with public health costs.

Truth disseminators significantly reduce rumor scale and cost accumulation.

Network structure modulates rumor control efficiency: Regular networks show smaller final rumor scales than BA scale-free networks at high truth spread rates.

4.2. Innovations. First coupling of rumor model with public health cost function for quantitative impact assessment;

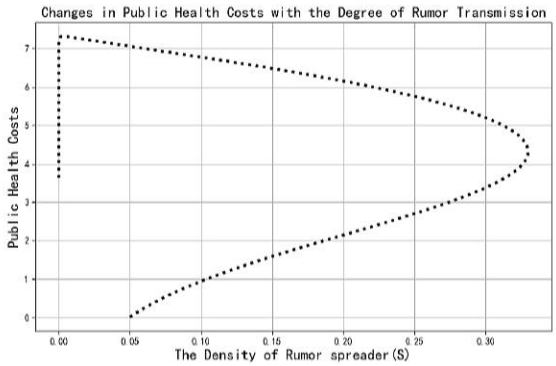


Figure 4. Changes in public health costs with the degree of rumor transmission

Revealed interaction mechanisms between rumor-refutation parameters and network structures.

4.3. Future research. Compare public health costs in ER random and scale-free networks and analyze impacts of parameter variations (e.g., λ, ε) on rumor severity.

References

1. Moreno Y., Nekovee M., Pacheco A. F. Dynamics of rumor spreading in complex networks // Physical Review E. 2004. Vol. 69. No 6. Art. no. 066130.

2. Qiu X., Wang J. A rumor spreading model with the truth-telling people in social networks // Journal of Technology. 2022. Vol. 22. No 4. P. 413–420. (In Chinese)

3. Wang X., Zhao L., Wu Z. Rumor propagation model with considering refutation mechanism in homogeneous network // Systems Engineering. 2015. Vol. 33. No 12. P. 139–145. (In Chinese)

4. Zhao L., Wang Q., Cheng J. et al. Rumor spreading model with consideration of forgetting mechanism: A case of online blogging LiveJournal // Physica A: Statistical Mechanics and its Applications. 2011. Vol. 390. Iss. 13. P. 2619–2625.

4. Информационные и компьютерные технологии



УДК 519.168

Авраменко П. А.

Приближенные раскраски графов в телекоммуникационных сетях

Рекомендовано к публикации доцентом Ереминым А. С

1. Введение. Раскраска графов имеет множество практических приложений, к примеру, распределение радиочастот, хранение химических веществ, составление расписаний, распределение регистров в микропроцессорах, политическая картография, окраска соединительных проводов. В данной работе раскраска используется для устранения конфликтов вышек связи. Целью работы ставится реализовать некоторые алгоритмы раскраски и провести их сравнительный анализ.

2. Граф. Поставленная задача, подразумевает наличие графа, имитирующего реальное расположение телекоммуникационных вышек и их связи. Связью будем считать пересечение покрытий двух вышек, каждому ребру будет соответствовать случайный вес в промежутке (0,1). Такой граф должен быть: а) разреженным; б) взвешенным; в) соответствовать заданным параметрам; г) имитировать расположение вышек на плоскости. Был реализован генератор графов, принимающий такие параметры, как количество вершин и средняя степень вершины. Случайные графы, полученные первой версией генератора, подходили для исследования алгоритмов раскраски, но не совсем соответствовали пункту г), поскольку любая вершина могла быть смежной с любой другой вершиной (кроме себя самой) (см. рис. 1).

В финальной версии генератора каждой вершине (вышке) сопоставлялись случайные координаты (x, y), полученные при помощи равномерного распределения, затем вершины связывали с k ближайшими соседями в определенном радиусе, где k – заданный параметр генератора. Таким образом мы получаем граф, который действительно похож на расположение вышек в черте города (см. рис. 2).

Авраменко Павел Александрович – студент, Санкт-Петербургский государственный университет; e-mail: avramenko.pav@yandex.ru, тел.: +7(916)554-85-48

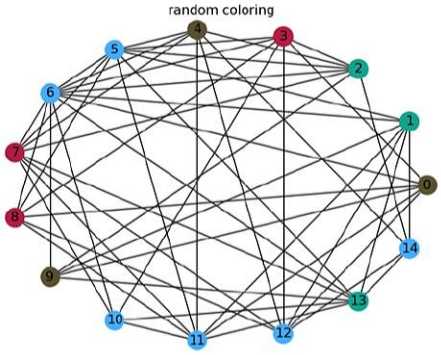


Рис. 1. Первая версия графа

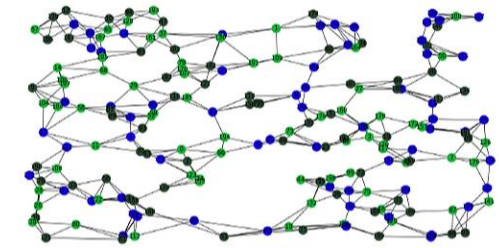


Рис. 2. Финальная версия генератора графов

3. Алгоритмы. Сопоставим вершинам графа (телекоммуникационным вышкам) конечное количество цветов (частоты, на которых они работают), конфликтом будем считать связь (пересечение покрытий) двух одноцветных вершин (вышек с одинаковыми частотами). Ошибкой будем называть сумму весов ребер, инцидентных вершинам одного цвета. Алгоритмы раскраски призваны минимизировать ошибку раскраски на графе.

3.1. Жадный алгоритм «восхождения на гору». Алгоритм «Восхождения на гору» – простой алгоритм нахождения локального оптимума хорошо показывает себя в задачах, где требуется поиск локального максимума или минимума, однако в представленной задаче лучшим исходом будет получить глобальный оптимум,

именно поэтому данный алгоритм показывает не лучшие результаты.

3.2. Алгоритм отжига. Метод имитации отжига является стохастическим алгоритмом, который позволяет «ошибаться» во время выполнения, то есть выбирать в конкретный момент времени не лучшее положение, что в последствии помогает избежать остановки на локальном оптимуме. В методе имитации отжига мы хотим минимизировать функцию потерь (итоговую ошибку). Нужно быть аккуратным с подбором входных параметров: начальной температурой и скоростью охлаждения, так как если охлаждение будет слишком быстрым, то существует вероятность попасть в локальный минимум, с другой стороны, если оно будет слишком медленным, будет невозможно достичь минимум за разумное время. Суть его работы: выбираем случайную вершину и случайный цвет, не равный нынешнему; если для этой вершины ошибка с новым цветом меньше, чем со старым (раскраска улучшилась) принимаем этот цвет, если раскраска не улучшилась мы принимаем новый цвет с некоторой вероятностью; уменьшаем температуру. Алгоритм продолжает работу до тех пор, пока температура не достигнет значения близкого к 0 [1].

3.3. Алгоритм G1. Данный алгоритм подходит исключительно для раскраски в 3 цвета и, даже при такой раскраске, серьезно проигрывает другим алгоритмам. Суть его работы: пока все вершины не раскрашены, выбираем не отмеченную вершину v с максимальным суммарным весом инцидентных ребер. Отмечаем вершину v и назначаем цвет 1. Переходим по не отмеченным вершинам, смежным с вершиной v, и раскрашиваем эти вершины цветами 2 и 3 и отмечаем их [2].

3.4. Алгоритм G2. Как и 2 последующих алгоритма, является детерминированным и показывает отличные результаты на исследуемых графах. Суть его работы: пока все вершины не окрашены, выполним следующие действия: для каждой неокрашенной вершины и для каждого цвета определим прирост целевой функции, если цвет присваивается вершине; выбираем вершину и цвет, который приведет к минимальному увеличению целевой функции и присваиваем этот цвет выбранной вершине [2].

3.5. Алгоритм G3. Детерминированный алгоритм со следующим шагами:

1. Отсортируем вершины в порядке не возрастания весов.
2. Для каждого ребра назначим цвет, соответствующий мини-

мальному вкладу целевой функции для каждой неокрашенной вершины ребра [2].

3.6. Алгоритм G4. Показывает лучшие результаты среди всех представленных алгоритмов.

1. Для каждой вершины вычисляем общий вес инцидентных ребер.
2. Отсортируем вершины в порядке убывания общего веса инцидентных им ребер.
3. Раскрасим вершины цветом, соответствующим минимальному вкладу целевой функции [2].

4. Эталон. В качестве эталона был взят линейный целочисленный минимизатор Gurobi, который получает решение близкое к глобальному оптимуму, однако тратит на решение крайне продолжительное время и требует большого количества вычислительных мощностей. Gurobi – зарубежная платная библиотека, которая не позволяет запускать решатель на задачах с более чем 2000 ограничений на бесплатной версии. Количество ограничений рассчитывается, как (кол-во ребер $\times$ кол-во цветов) + кол-во вершин. Для целевого графа 2000 вершин, средней степенью вершины 20 и 5 цветами получим 202000 ограничений, что в 101 раз больше, чем допустимо. Решением данной проблемы стала кластеризация графа, т. е. его разбиение на связные подграфы, которые доступны для решения задачи. Однако при таком подходе мы жертвуем ребрами, связывающими кластеры, что отдаляет нас от идеальной раскраски. Такой подход имеет свои недостатки, однако дает сравнительную базу для исследования работы методов.

5. Сравнительный анализ. Изначально целью исследования ставилось подобрать такие входные параметры для метода отжига, чтобы он показал лучшие результаты, чем остальные методы. Будем исследовать алгоритмы на графах от 40 до 500 вершин и средней степенью вершин 15 (см. рис. 3).

Как можно заметить, с такими параметрами алгоритм отжига показывает не лучшие результаты по сравнению с другими. Эмпирически было выяснено, что повышение начальной температуры выше 10 не дает значительного вклада в результат, поэтому остановимся этом значении (см. рис. 4).

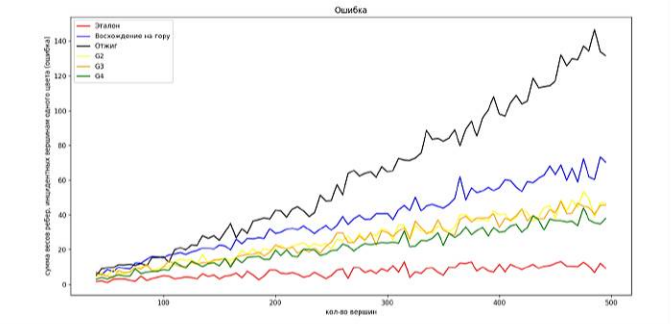


Рис. 3. 5 цветов, стартовая температура: 1.25; скорость убывания: 1.01

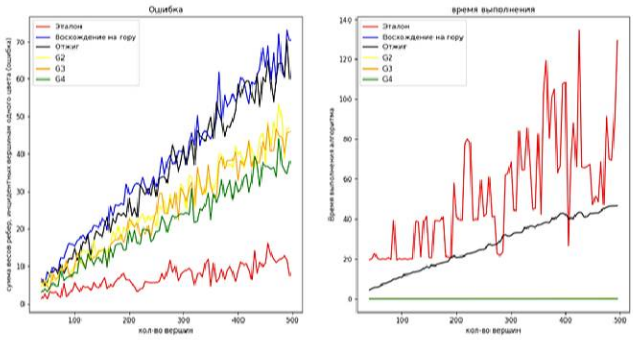


Рис. 4. 5 цветов, стартовая температура: 10; скорость убывания: 1.001

Последующий анализ показал, что алгоритм отжига может приблизиться к результатам алгоритмов G2, G3 и G4, за неразумное время, превышающее время работы эталона (см. рис. 5).

6. Вывод. Детерминированные алгоритмы показывают лучший результат на поставленной задаче. Вероятно, для алгоритма отжига можно задать такие параметры, чтобы он превосходил другие представленные алгоритмы, однако задача не будет решаться за разумное время. Однако у алгоритмов есть пространство, для улучшение, возможно в новой конфигурации алгоритм отжига сможет превзойти детерминированные алгоритмы как по результату, так и по времени работы.

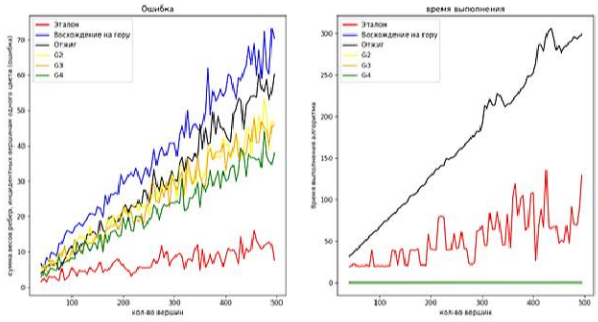


Рис. 5. 5 цветов, стартовая температура: 10; скорость убывания: 11.0001

Литература

1. Kose A., Sonmez B. A., Balaban M. Simulated annealing algorithm for graph coloring //arXiv preprint arXiv:1712.00709. 2017.

2. Erzin, A., Plotnikov, R., Zhukov, G. Minimizing the sum of weights of single-color edges in a 3-coloring. // Communications in Computer and Information Science. 2024.

УДК 004.85

Аржа А. Н.

Применение автокодировщиков для поиска аномальных временных рядов в наборе данных с умных счетчиков

Рекомендовано к публикации доцентом Корховым В. В.

1. Введение. Данные в виде временных рядов встречаются в самых разных областях – в энергетике [1], финансах [2], медицине [3], производстве [4] и других. Каждая из этих сфер в значительной степени зависит от способности обнаруживать аномалии, чтобы снизить риски или оптимизировать процессы.

Большинство методов поиска аномалий во временных рядах можно отнести к одному из семи семейств: классическое машинное обучение, анализ сигналов, обнаружение выбросов, стохастическое обучение, статистика, интеллектуальный анализ данных и глубокое обучение. Несмотря на разнообразие существующих state-of-the-art алгоритмов, решающих данную задачу, большинство их них ненадежны, трудны в настройке или, несмотря на свою сложность, не имеют выдающегося превосходства над более простыми методами [5].

Задача поиска аномальных временных рядов из набора – одна из постановок проблемы обнаружения аномалий во временных рядах. В данной постановке требуется рассматривать временные ряды и судить об их аномальности, опираясь на набор временных рядов, а не на один конкретный экземпляр. Такая проблема возникает, например, при обработке данных с умных счетчиков (электроэнергии и т. п.). Традиционные методы, ориентированные на анализ одиночных временных рядов, часто оказываются неприменимыми без модификаций, что требует разработки специализированных подходов.

В связи с тем, что ручная разметка больших объемов данных зачастую является практически невыполнимой задачей, большой интерес представляют подходы, относящиеся к парадигме обучения без учителя, т. е. не требующие размеченной выборки с заранее выделенными аномалиями [6].

Аржа Антон Насерович – студент, Санкт-Петербургский государственный университет; e-mail: st102714@student.spbu.ru, тел.: +7(981)747-54-00

В настоящее время нейронные сети широко применяются для решения различных задач, в том числе и задачи поиска аномалий во временных рядах [7]. Одной из архитектур нейронных сетей, решающих задачи обучения без учителя, являются автокодировщики.

В данной работе рассмотрены несколько подходов для обнаружения аномальных временных рядов, основанные на автокодировщиках. Их результаты проанализированы на примере набора данных умных счетчиков электроэнергии UK Power Networks [8]. Данные и исходный код моделей доступны в репозитории [9].

2. Описание и подготовка данных. Выбранный набор данных содержит информацию о показаниях потребления различных низковольтных автоматов (LV feeder), которые для краткости будут называться устройствами. Данные представлены в виде измерений каждые полчаса на протяжении месяца.

Для дальнейшего анализа из имеющихся показателей был выбран один, отражающий активное (полезное) потребление электроэнергии. Далее были выделены только те устройства, для которых в данных нет пропусков, так как их заполнение является отдельной нетривиальной задачей. Затем данные были сгруппированы в итоговую матрицу, где строки соответствуют различным устройствам, а столбцы – временным меткам. Дополнительно строки матрицы были нормализованы, что уменьшило время обучения нейросетей.

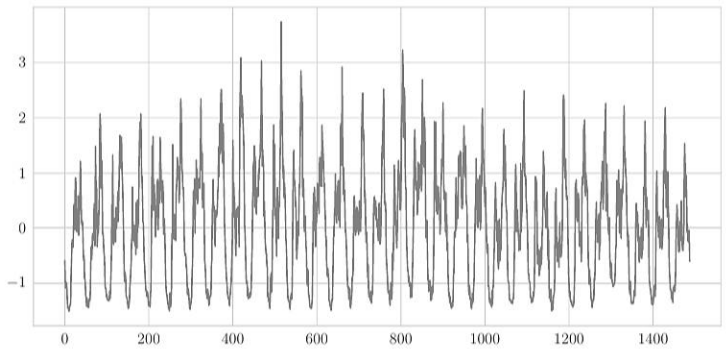


Рис. 1. Пример временного ряда

На рис. 1 изображен график типичного временного ряда. Большинство устройств в наборе данных обладают схожими кривыми потребления, в которых заметна четкая дневная периодичность.

3. Модели, основанные на автокодировщиках. Автокодировщик – это нейронная сеть, состоящая из двух подсетей, имеющих зеркальную структуру. Первая из них – кодировщик, сжимает исходные данные в некоторое скрытое представление, тогда как вторая – декодировщик, восстанавливает из него исходные данные. Так как размерность кодирования меньше исходной, тривиальные преобразования невозможны. Поэтому, если нейронная сеть обучилась восстанавливать данные с небольшой ошибкой, это говорит о том, что на выходе кодировщика получается их сжатое представление.

Опираясь на некоторые предположения, можно построить модели выявления аномальных временных рядов, основанные на автокодировщике. Во-первых, предположив, что обученная модель восстанавливает нормальные данные лучше, чем аномальные, можно использовать ошибку восстановления временного ряда в качестве показателя, характеризующего его аномальность. Во-вторых, размерность скрытого представления может быть значительно меньше исходной. Тогда предположив, что аномальные последовательности кодируются аномальными параметрами, можно провести кластеризацию закодированных временных рядов. Например, алгоритм DBSCAN, представленный в [10], обладает естественной способностью отмечать как аномалии точки данных, не вошедшие ни в один кластер.

Автокодировщик можно построить на основе различных архитектур нейросетей. Компоненты классического варианта являются полносвязными сетями. Для работы с последовательностями также применяются сверточные сети (CNN), рекуррентные сети (RNN) и трансформеры. В данной статье рассматриваются автокодировщики всех этих видов и методы обнаружения аномальных временных рядов, полагающиеся на сформулированные выше предположения.

4. Реализованные модели. Для дальнейших экспериментов были обучены классический (AE), сверточный (CAE), рекуррентный (RAE) и трансформерный (TAE) автокодировщики.

Модели сжимали исходные данные более чем в 30 раз, преобразуя исходный вектор, состоящий из почти полутора тысяч наблюдений, в 48-мерное закодированное представление.

Для большей объективности сравнения архитектура моделей была построена таким образом, чтобы общее количество обучаемых параметров было примерно равным и составляло около полумиллиона. Исходная выборка сначала делилась на обучающую и тестовую, чтобы найти оптимальное количество эпох и избежать переобучения. Затем модели обучались с нуля на полной выборке.

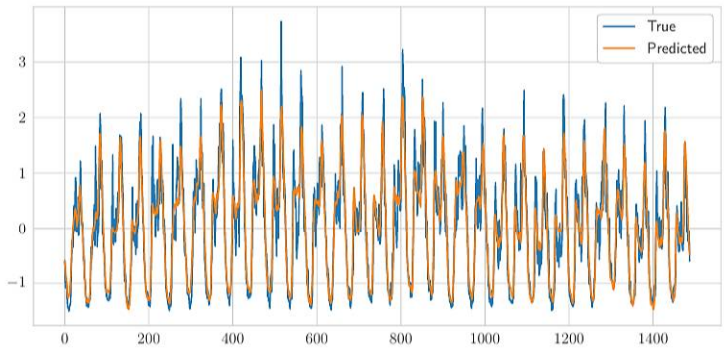


Рис. 2. Пример восстановленного временного ряда

Все модели, в среднем, с высокой точностью обучились восстанавливать исходные данные. Примечательно, что несмотря на разную архитектуру и количество эпох, в среднем точность получилась примерно одинаковая у различных автокодировщиков. На рис. 2 представлен пример восстановления временного ряда с помощью классического автокодировщика.

Далее аномальные временные ряды выделялись двумя способами: по ошибке восстановления и с помощью кластеризации. Параметры алгоритма DBSCAN подбирались таким образом, чтобы определенное количество рядов было отмечено аномальными.

5. Результаты. Результаты работы реализованных методов были проанализированы с целью сравнения различных моделей между собой. Интерес представляет то, какие виды аномалий выявляет каждый подход.

В первую очередь, модели были сравнены по времени, затраченному на обучение.

Таблица. Время обучения моделей

Модель	Время обучения, сек
AE	31
CAE	14
RAE	107
TAE	24

Из данных, представленных в таблице, видно, что быстрее всего обучилась сверточная модель. Немного отстают от нее трансформерная и классическая. Гораздо большее время было затрачено на обучение рекуррентной сети.

В дальнейшем сравнивались топ-10 аномалии, обнаруженные каждым из восьми способов. Примечательно, что значительная часть (44%) аномалий идентифицируется сразу несколькими (тремя и более) способами, что говорит о согласованности результатов. Напротив, для одной архитектуры, аномалии, обнаруженные с помощью ошибки восстановления и кластеризации, не совпадают. В целом, подход, основанный на ошибке восстановления, отмечает как аномалии временные ряды с необычными скачками потребления или общей зашумленностью. С другой стороны, подход, основанный на кластеризации, чаще выделяет аномалии, характеризующиеся необычными по форме кривыми потребления. Некоторые временные ряды имеют нулевые участки, что также выявляется этим подходом. На рис. 3–5 приведены примеры разных видов аномалий, обнаруженных с помощью реализованных моделей. В заголовках этих рисунков в скобках указаны модели, графики предсказаний которых наложены на исходные данные.

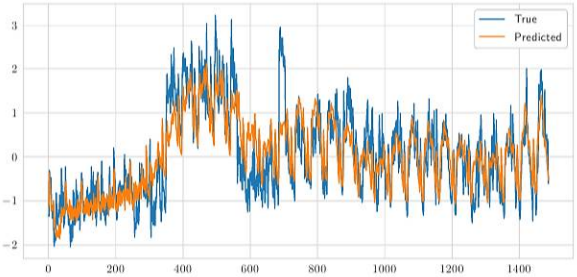


Рис. 3. Временной ряд с необычной кривой потребления (RAE)

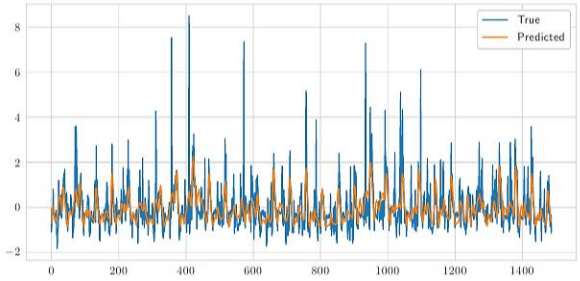


Рис. 4. Временной ряд с необычными пиками (CAE)

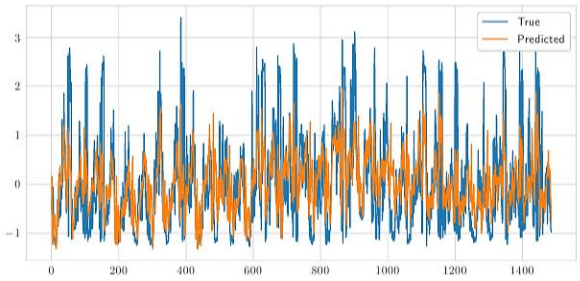


Рис. 5. Временной ряд с необычной зашумленностью (AE)

Дополнительно следует отметить, что результаты классического автокодировщика меньше всего согласуются с другими моделями, а также он хуже выделяет аномальные кривые потребления.

6. Заключение. В работе рассмотрены подходы к обнаружению аномальных временных рядов с использованием автокодировщиков различных архитектур. С помощью реализованных моделей можно обнаруживать различные виды аномалий.

Подход, основанный на ошибке восстановления, позволяет выявлять временные ряды с нетипичными скачками или зашумленностью, тогда как применение кластеризации позволяет идентифицировать аномальные кривые потребления.

Среди рассмотренных моделей предпочтительными являются CAE и TAE, так как они лишены недостатков моделей AE и RAE.

Литература

1. Zhang J. E., Wu D., Boulet B. Time series anomaly detection for smart grids: a survey // 2021 IEEE Electrical Power and Energy Conference (EPEC). 2021. P. 125–130.
2. Guo C. M., Xu L. Y., Liu H. F., Wang L., Yu X., Han B. The financial data of anomaly detection research based on time series // 2015 International Conference on Computer Science and Applications (CSA). 2015. P. 86–89.
3. Легкова И. А. Обнаружение отклонений и нарушений сердечного ритма в данных ЭКГ пациентов // Процессы управления и устойчивость. 2024. Т. 11. № 1. С. 150–154.
4. Theumer P., Zeiser R., Trauner L., Reinhart G. Anomaly detection on industrial time series for retaining energy efficiency // Procedia CIRP. 2021. Vol. 99. P. 33–38.
5. Schmidl S., Wenig P. Anomaly detection in time series: a comprehensive evaluation // Proceedings of the VLDB Endowment. 2022. Vol. 15. No 9. P. 1779–1797.
6. Belay M. A., Blakseth S. S., Rasheed A., Salvo Rossi P. Unsupervised anomaly detection for IoT-based multivariate time series: existing solutions, performance analysis and future directions // Sensors. 2023. Vol. 23. No 5. Art. no 2844.
7. Li G., Jung J. J. Deep learning for anomaly detection in multivariate time series: approaches, applications, and challenges // Information Fusion. 2023. Vol. 91. P. 93–102.
8. UK Power Networks [Электронный ресурс]: `URL:https://ukpowernetworks.opendatasoft.com/` (дата обращения: 26.02.25).
9. Репозиторий GitHub [Электронный ресурс]: `URL:https://github.com/UnicornTowa/MTSADAE/` (дата обращения: 26.02.25).
10. Ester M., Kriegel H.-P., Sander J., Xu X. A density-based algorithm for discovering clusters in large spatial databases with noise // Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD'96). 1996. P. 226–231.

УДК 004.912

Аристакесян Т. А., Блеканов И. С.

Методы декомпозиции тем пользовательской дискуссии на основе динамического тематического моделирования

1. Введение. В условиях массового перехода к повседневному сетевому общению социальные дискуссии в медиаплатформах, таких как Телеграм, стали ключевым источником для анализа общественных настроений, выявления тенденций и острых социальных проблем. Динамическое моделирование тем [1], направленное на отслеживание эволюции ключевых тем во времени, позволяет исследователям фиксировать сдвиги в фокусе внимания аудитории, реакцию на события и зарождение новых дискуссий. Однако классические методы, такие как LDA, сталкиваются с принципиальными ограничениями: они игнорируют временную динамику и семантическую многозначность текстов, что делает их непригодными для анализа нелинейных процессов – ветвления тем на подтемы, их слияния или резкого исчезновения.

2. Проблематика. Анализ социальных дискуссий в социальных медиа осложняется рядом проблем, включая динамику изменений в теме, семантическую неоднозначность, изменчивость обсуждений и языковую специфику. Темы эволюционируют непредсказуемо, что делает статические модели неэффективными. Семантическая многозначность затрудняет кластеризацию, так как одни и те же слова могут менять значение в зависимости от контекста. Дискуссии часто дробятся на подкатегории, требуя отслеживания структурных изменений. Эти проблемы особенно актуальны для русского языка, учитывая его сложную морфологию, синтаксическую гибкость и высокую степень омонимии. В данной работе исследуется алгоритм ди-

Аристакесян Тигран Арменович – студент, Санкт-Петербургский государственный университет; e-mail: st097581@student.spbu.ru, тел.: +7(993)221-18-53

Блеканов Иван Станиславович – доцент, Санкт-Петербургский государственный университет; e-mail: i.blekanov@spbu.ru, тел.: +7(812) 363-65-37

Исследование проведено в рамках проекта «Накопительное общественное мнение и кумулятивная делиберация: теоретизация, методы анализа и оценка качества современного делиберативного процесса», ID 128786104, 2025

намического моделирования тем, адаптированный для анализа русскоязычных дискуссий в социальных сетях.

3. Предлагаемые методы. В работе была разработана общая схема работы алгоритмов декомпозиции тем дискуссии, основанная на динамическом тематическом моделировании, которая включает следующие модули: (см. рис. 1):

- модуль модели BERTopic [2] в связке с большой языковой моделью для формирования всех топиков в документах;
- модуль репрезентации тем;
- модуль поиска оптимального размера интервала для каждой отдельной темы;
- модуль представления тем в интервалах;
- алгоритм ветвления тем;
- модуль визуализации.

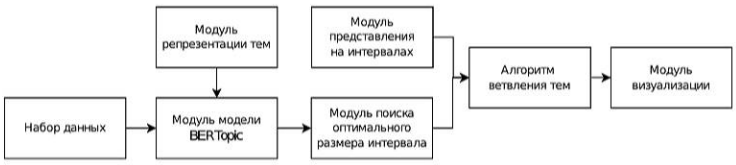


Рис. 1. Схема работы алгоритмов декомпозиции тем дискуссии

4. Методология и архитектура. В данном разделе приводится детальное описание методологических подходов и архитектурных решений, лежащих в основе разработанной системы декомпозиции тем.

4.1. Модуль модели BERTopic. Перед построением начальной модели BERTopic необходимо выбрать конфигурацию этой модели, а именно: определить подходящую модель для построения эмбеддингов, понижения размерности, кластеризации, а также репрезентации. Для качественного и эффективного построения модели рекомендуется использовать современные LLM (большие языковые модели), которые способны учитывать контекст высказываний, семантические связи между словами и специфику естественного языка.

4.2. Модуль репрезентации тем. При работе с русскими текстами нужно учитывать их особенности – стоп-слова и форма слова. Для этого была разработана кастомная модель репрезентации тем, в которой стоп-слова удаляются, а остальные слова приводятся к нормальной форме, что делает итоговое представление тем более точным и информативным. Важным аспектом является то, что модуль репрезентации никак не влияет на само построение тем, а лишь на их конечное представление.

4.3. Модуль поиска оптимального размера интервала. Под оптимизацией понимается динамическая сегментация «времени жизни» темы на интервалы, где длина окна подбирается таким образом, чтобы максимизировать среднюю согласованность внутри всех окон. Метрика согласованности количественно оценивает, насколько темы внутри временного интервала являются семантически однородными. Таким образом, после нахождения оптимального интервала, тема разделяется на N максимально взаимосвязанных частей. Алгоритм представляет собой перебор всех натуральных чисел N от 2 до «времени жизни» темы и разбиения на эти N интервалов с вычислением средней согласованности. Перебрав все варианты, находим наилучшее разбиение.

4.4. Модуль представления тем в интервалах. Для дальнейшего анализа отдельных интервалов нужно получить репрезентацию каждого из них – слова, описывающие тему на каждом промежутке. Строим репрезентацию для каждого интервала с помощью c-TF-IDF [3], а также на каждом интервале учитываем представление глобальной темы и представление тем на интервалах, которые шли до текущего. Для этого нормализуем c-TF-IDF представления глобальной темы и текущего интервала и возьмем среднее суммы двух распределений. Аналогичную технику применяем к предыдущим интервалам.

4.5. Алгоритм ветвления тем. Ниже представлен алгоритм ветвления темы дискуссии. Класс `Branch` – ветка, у который есть начало, конец и датафрейм, который содержит все интервалы с документами, относящиеся только к данной ветке.

1. Инициализация:
   - создать начальную ветку `Branch` с начальным интервалом, содержащим все документы;
   - добавить начальную ветку в список `active_branches`, содержащий все активные ветки.

2. Цикл обработки веток (повторять пока `active_branches` не пусто):
    - взять текущую ветку `current_branch` из списка `active_branches`;
    - если наступила точка бифуркации:
        - зафиксировать точку бифуркации;
        - разделить интервалы до точки бифуркации и после;
        - применить BERTopic с новой конфигурацией для интервалов после точки;
        - на основе нового распределения создать новые ветки new_branch для интервалов после точки бифуркации;
        - добавить `new_branch` в список `active_branches`.

4.6. Модуль визуализации. После алгоритма ветвления тем получаем граф, который описывает, как разные ветки в одной теме взаимосвязаны. На основе этого графа с использованием библиотеки Python networkx [4] строим визуализацию, которая учитывает расположение веток относительно друг друга и очередность интервалов. На графе каждая вершина отвечает за интервал, и выводится его репрезентация. В итоге получаем инструмент для удобного анализа темы, ее эволюции и ветвления. Пример ветвления темы, полученный алгоритмом «stsb-xlm-r-multilingual + Isomap + K-means», представлен на рис. 2. На нём изображено ветвление темы «школа – образование – ребенок – учитель», на втором и третьем интервале найдены точки бифуркации и после них произошло разветвление на три подтемы.

5. Эксперимент. В работе ставился эксперимент по оценке качества четырех алгоритмов декомпозиции темы с разными конфигурациями BERTopic:

1. paraphrase-multilingual-MiniLM-L12-v2 + UMAP + HDBSCAN
2. stsb-xlm-r-multilingual + UMAP + HDBSCAN
3. paraphrase-multilingual-MiniLM-L12-v2 + Isomap + K-means
4. stsb-xlm-r-multilingual 278M + Isomap + K-means

Оценка производилась по двум метрикам: согласованность всех тем дискуссии и средняя согласованность в теме после оптимального разделения на интервалы.

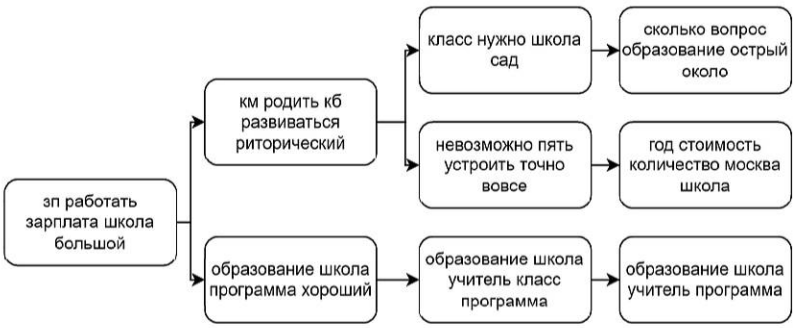


Рис. 2. Ветвление темы «школа – образование – ребенок – учитель» после пайплана «stsb-xlm-r-multilingual 278M + Isomap + K-means»

5.1. Датасет. Набор данных для тестирования построен на основе API Telegram [5], включающий посты и комментарии из российских Телеграм-каналов, собранные за период с 17 февраля по 2 марта 2025 года. Набор данных содержит 230 тыс. документов, после первичной обработки осталось 115 тыс. Убраны все документы, которые содержат менее шести слов. Это сделано, чтобы повысить информативность коллекции документов и улучшить качество тематического моделирования. Для демонстрации работы алгоритма отобрано четыре наиболее релевантных темы из всей коллекции документов, имеющие следующие представления:

1. «зарплата – работа – платить – работать»;
2. «школа – образование – ребенок – учитель»;
3. «полиция – полицейский – стрелять – закон»;
4. «врач – мед – больница».

5.2. Результаты. В таблице 1 представлены результаты для всех конфигураций по метрике согласованности для всех тем. В таблицах 2–5 представлены показатели средней согласованности по темам дискуссии для четырех конфигураций. Наилучший результат получен в конфигурацией «stsb-xlm-r-multilingual + Isomap + K-means»

Таблица 1. Результаты по конфигурациям

Конфигурация	Согласованность
1. paraphrase-multilingual-MiniLM-L12-v2 + UMAP + HDBSCAN	0,71
2. stsb-xlm-r-multilingual+ UMAP + HDBSCAN	0,73
3. paraphrase-multilingual-MiniLM-L12-v2 + Isomap + K-means	0,72
4. stsb-xlm-r-multilingual + Isomap + K-means	0,71

Таблица 2. Результаты по темам для конфигурации 1

Тема	Кол-во интервалов	Средняя согласованность
0	9	0,386
1	6	0,645
2	12	0,648
3	4	0,909

Таблица 3. Результаты по темам для конфигурации 2

Тема	Кол-во интервалов	Средняя согласованность
0	9	0,439
1	6	0,691
2	12	0,586
3	4	0,911

Таблица 4. Результаты по темам для конфигурации 3

Тема	Кол-во интервалов	Средняя согласованность
0	9	0,398
1	6	0,667
2	12	0,653
3	4	0,909

Таблица 5. Результаты по темам для конфигурации 4

Тема	Кол-во интервалов	Средняя согласованность
0	9	0,444
1	6	0,703
2	11	0,540
3	4	0,911

6. Заключение. Предложенные в работе алгоритмы декомпозиции тем пользовательской дискуссии на основе динамического тематического моделирования с конфигурацией «Isomap + K-means» продемонстрировали превосходство в качестве результатов по сравнению с парой «UMAP + HDBSCAN» при использовании идентичных больших языковых моделей. Кроме того, в исследовании обнаружена прямая зависимость: с увеличением количества параметров в модели, используемой для построения эмбеддингов, наблюдается рост их согласованности. В дальнейшем планируется расширить эксперимент с добавлением методов абстрактивной суммаризации для повышения интерпретируемости полученных декомпозированных тем дискуссии.

Литература

1. Blekanov I. S., Tarasov N., Bodrunova S. S., Sergeev S. L. Mapping opinion cumulation: topic modeling-based dynamic summarization of user discussions on social networks // In: Coman A., Vasilache S. (eds) Social Computing and Social Media. HCII 2023. Lecture Notes in Computer Science. Springer, Cham. Vol. 14025. P. 25–40.

2. BERTopic [Электронный ресурс]: URL: `https://github.com/MaartenGr/BERTopic?tab=readme-ov-file` (дата обращения: 05.03.25).

3. c-TF-IDF [Электронный ресурс]: URL: `https://github.com/MaartenGr/cTFIDF` (дата обращения: 17.03.25).

4. Networks – библиотека Python для визуализации графов [Электронный ресурс]: URL: `https://networkx.org/` (дата обращения: 02.02.25).

5. Telegram API [Электронный ресурс]: URL: `https://core.telegram.org/` (дата обращения: 28.01.25).

УДК 004.932.72.1

Бабкина А. В., Вахитов Б. Д., Девришев Т. Э., Мишин И. В., Петренко А. А.

Детектирование признаков туберкулеза на рентген-флюорографиях легких

Рекомендовано к публикации доцентом Блекановым И. С.

1. Введение. Рентген-флюорография является методом исследования внутренних структур органов, основанным на прохождении рентгеновских лучей сквозь тело человека с целью получения суммарного проекционного изображения. Анализ рентгеновских снимков – ключевой этап диагностики заболеваний легких, требующий от врача высокого уровня теоретических знаний, практических навыков, опыта и концентрации внимания. В условиях высокой нагрузки на медицинский персонал [1] возрастает риск диагностических ошибок, а позднее выявление патологий (например, туберкулеза) приводит к осложнениям и увеличению смертности.

Применение технических решений, базирующихся на использовании искусственного интеллекта на основе нейронный сетей глубокого обучения, позволяет ускорить и оптимизировать анализ рентгеновских снимков, снижая вероятность ошибок и облегчая нагрузку на врачей. Нейросеть не заменяет специалиста, но может ускорить постановку предварительного диагноза, отметив на снимке подозрительные артефакты, на которые, возможно, стоит обратить внимание.

При разработке алгоритма детектирования заболеваний легких важным требованием является высокая точность детектирования

Бабкина Анна Владимировна – магистрант, Санкт-Петербургский государственный университет; e-mail: st076184@student.spbu.ru, тел.: +7(904)331-72-88

Вахитов Булат Дамирович – магистрант, Санкт-Петербургский государственный университет; e-mail: st075834@student.spbu.ru, тел.: +7(986)909-10-44

Девришев Турал Эльнурович – студент, Санкт-Петербургский государственный университет; e-mail: st107569@student.spbu.ru, тел.: +7(939)716-76-50

Мишин Иван Владимирович – магистрант, Тульский педагогический университет им. Л.Н. Толстого; e-mail: imisin57@gmail.com, тел.: +7(920)757-13-85

Петренко Алексей Анатольевич – магистрант, Санкт-Петербургский государственный университет; e-mail: st070886@student.spbu.ru, тел.: +7(911)121-90-93

и интерпретируемость результатов. Модель должна предоставлять врачам понятную информацию, которая позволяет эффективно использовать ее для диагностики заболеваний. На выходе алгоритма необходимо получить классификацию снимка (есть туберкулез или нет), а также локализовать рамкой обнаруженные очаги поражения. Важно учитывать такие ограничения, как качество изображений, различные стандарты съемки и возможные артефакты на снимках, вызванные другими болезнями, а также недостаток или неполнота обучающих данных, что может существенно повлиять на точность работы модели.

В данной статье проведен сравнительный анализ двух архитектур нейронных сетей для решения задачи детектирования туберкулеза на рентгенограммах легких.

2. Обзор известных нейросетевых архитектур. Современные исследования в области анализа медицинских изображений демонстрируют прогресс в задачах сегментации и классификации различных патологий, включая анализ рентгенограмм легких. Например, работы, изначально ориентированные на сегментацию полипов желудочно-кишечного тракта, предлагают решения, подходящие для задач детектирования заболеваний легких. В исследовании [2] предложена адаптивная функция потерь Adaptive t-vMF Dice Loss, которая улучшает метрику Dice на датасетах CVC-ClinicDB и Kvasir-SEG, что критично для точного выделения границ патологий. Архитектурные инновации, такие как DUCK-Net [3] с комбинацией различных типов сверток, а также гибридная модель RAPUnet [4] (трансформеры + CNN), достигли наилучших на данный момент результатов в сегментации полипов, подчеркивая важность сочетания локальных и глобальных признаков. Перечисленные методы могут быть адаптированы для анализа легких, где требуется учет как мелких деталей, так и общего контекста. Эффективность применения переноса обучения продемонстрирована в работе [5], где предобученная EfficientNet обеспечила близкие к наилучшим результаты на наборе данных Kvasir-SEG, что особенно актуально для медицинских задач с ограниченными данными.

В задачах, непосредственно связанных с легкими, ключевыми остаются архитектуры U-Net и DenseNet. Так, сеть на основе U-Net с функцией потерь бинарная кросс энтропия (BCE) + Jaccard [6] показала высокую эффективность для сегментации снимков легких на

платформе Kaggle, но интеграция адаптивных функций потерь, например, из работы [2], вероятно может повысить точность. Для классификации пневмонии комбинация DenseNet-169 и SVM [7] достигла высокой точности на наборе данных ChestX-ray14, а сеть CheXNet [8] на основе 121-слойной DenseNet не только классифицирует 14 заболеваний, но и визуализирует области интереса через тепловые карты, усиливая интерпретируемость диагноза.

Дальнейшее улучшение результатов, предположительно, может быть связано с переносом описанных методов (гибридные архитектуры [4], адаптивные функции потерь [2] и предобученные модели [5]) в область анализа рентгенограмм легких. Однако основной трудностью является адаптация данных подходов к специфике рентгенограмм легких, включая низкий контраст изображений и наложение других анатомических структур. Таким образом, существующие исследования создают основу для комплексных решений задач детекции заболеваний легких, но требуют доработки и оптимизации.

3. Предлагаемые нейросетевые архитектуры. Ниже представлены две исследуемые в работе архитектуры.

3.1. YOLO. Сеть YOLOv11 Nano (рис. 1), разработанная Ultralytics, представляет собой компактную версию популярной архитектуры YOLO для детектирования объектов, оптимизированную для устройств с ограниченными ресурсами. Ее трехкомпонентная структура включает базовую сеть (Backbone), отвечающую за извлечение признаков с помощью облегченных сверточных слоев, промежуточный модуль (Neck), собирающий признаки на разных масштабах изображения для детектирования объектов любого размера, и выходной модуль (Head), генерирующий предсказания координат рамок и вероятности классов.

Для обучения модели использовалась комбинированная функция потерь, объединяющая среднеквадратичную ошибку (MSE) для локализации рамок, бинарную кросс-энтропию (BCE) для классификации объектов и оценку уверенности в их наличии. Модель содержала 2,5 млн параметров, что в 4 раза меньше, чем у YOLOv8n, сохраняя при этом высокую точность.

3.2. DETR Архитектура DETR [10] представлена на рис. 2. Она содержит три основных компонента: слой backbone, включающий в себя сверточную сеть для извлечения компактного представления признаков, трансформер, состоящий из encoder-decoder блоков и набора сетей прямого распространения (feed forward network – FFN),

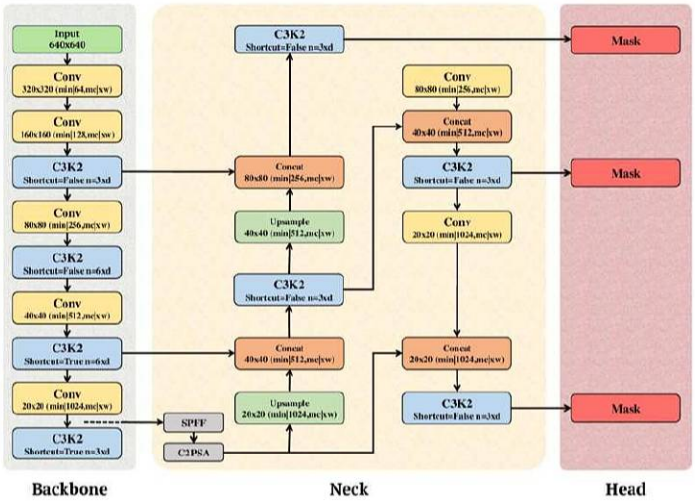


Рис. 1. Архитектура сети YOLOv11 [9]

дающую окончательный прогноз обнаружения.

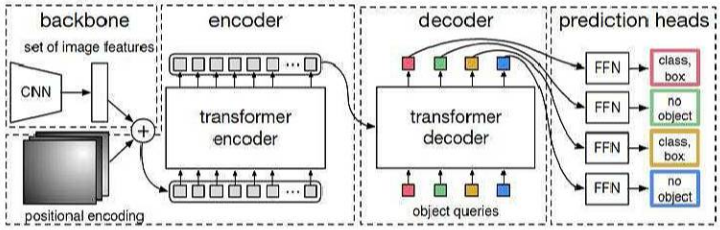


Рис. 2. Архитектура сети DETR [10]

Для обучения сети DETR применялась комбинированная функция потерь, объединяющая три ключевых компонента: кросс-энтропию для классификации объектов, L1 для точного позиционирования центров и размеров ограничивающих рамок и метрику GIoU (Generalized Intersection over Union), оптимизирующую перекрытие предсказанных и целевых боксов. Модель содержала 41 млн параметров.

4. Наборы данных для обучения моделей. В настоящей работе рассмотрены следующие наборы данных:

1. TBX11K Simplified [11] содержит более чем 11 тыс. рентгеновских снимков с метками областей, пораженных туберкулезом.
2. NIH Chest X-rays (ChestX-ray14) [12] включает более 112 тыс. рентгеновских снимков, охватывающих 14 заболеваний грудной клетки. Данный набор полезен для обучения моделей, предназначенных для обнаружения нескольких видов заболеваний легких.
3. VinBigData Chest X-ray Abnormalities Detection [13] состоит из 18 тыс. рентгеновских снимков с аннотациями, выполненными сертифицированными рентгенологами.

Для дальнейшей работы был выбран набор TBX11K Simplified, как единственный набор данных, содержащий изображения легких с туберкулезом, отмеченным рамками. Данные были разбиты на обучающую, проверочную и тестовую выборки в отношении 8:1:1.

5. Эксперимент. В работе был поставлен эксперимент по сравнению результатов обучения двух моделей для решения задачи детектирования туберкулеза на рентгенограммах легких. Производилось сравнение показателей точности положений рамок (IoU), классификации заболевания (precision, recall, F1) и скорости обработки изображений. Результаты обучения и тестирования сетей приведены в таблице.

Обучение модели YOLOv11 проводилось на 500 эпохах с размером батча 32, что заняло 4,2 часа на GPU NVIDIA RTX4090.

Дообучение модели DETR проводилось на 30 эпохах с размером батча 8, что заняло 5 часов в Google Colab на GPU.

На тестовом наборе данных обе сети продемонстрировали высокие показатели метрик классификации precision и recall. Сеть DETR показала результаты лучше, чем YOLOv11 по метрике precision (0,97), однако уступила ей в значении recall и в точности детектирования границ рамок. Среднее значение IoU = 0,709 сети YOLOv11 говорит о достаточно точной локализации патологических изменений, а IoU = 0,56 сети DETR указывает на удовлетворительную точность и возможно требует доработки или дообучения.

Таблица. Сравнение моделей

	YOLOv11	**DETR**
Размер модели (количество параметров)	2,5 млн	41 млн
Точность положения рамки (IoU)	0,71	0,56
Точность классификации	precision = 0,95 recall = 0,95 F1 = 0,95	precision = 0,97 recall = 0,8 F1 = 0,87
Время обработки одного изображения (GPU)	30–50 ms Nvidia RTX4090	80–100 ms Google Colab

На рис. 3 показаны положения целевых и предсказанных рамок, ограничивающих область поражения туберкулезом, для сети YOLOv11 и DETR. В случае отсутствия патологии на изображении обе модели выделяют рамкой с пометкой “No finding” все изображение.

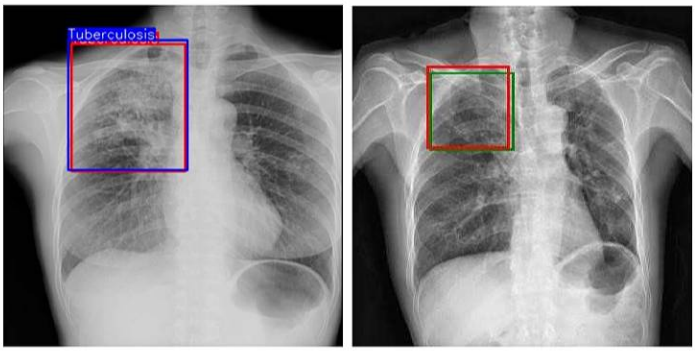


Рис. 3. Целевые и предсказанные рамки для сети YOLOv11 (слева) и сети DETR (справа)

6. Заключение. Применение существующих архитектур нейронных сетей продемонстрировало эффективность в задачах классификации и локализации патологии туберкулеза на рентгеновских снимках легких. В ходе исследования было выявлено, что использование сети YOLOv11 предпочтительнее, чем сети DETR, по показателям метрики классификации recall и метрики IoU детектирования положения рамки вокруг выявленной патологии. Существенным преимуществом YOLOv11 также является на порядок меньшая сложность и простота обучения.

В перспективе возможно совершенствование алгоритма для увеличения количества классов определяемых патологий и нескольких рамок на одном изображении. Необходимо также исследовать границы применимости данных методов на других наборах рентгеновских данных.

Литература

1. Dobretsova E. A., Arshukova I. L. et al. Shortage of doctors in remote and sparsely populated areas: ways to improve the situation // Вестник Санкт-Петербургского университета. Медицина. 2022. Т. 17. № 1. С. 43–52.
2. Kato S., Hotta K. Adaptive t-vMF Dice Loss for Multi-class Medical Image Segmentation [Электронный ресурс]: URL:https://arxiv.org/abs/2207.07842 (дата обращения 09.03.25)
3. Dumitru R. G., Peteleaza D., Craciun C. Using DUCK-Net for polyp image segmentation // Scientific Reports. 2023. Vol. 13. No 1. P. 9803.
4. Lee H., Yoo J. MetaFormer and CNN hybrid model for polyp image segmentation // IEEE Access. 2024. Vol. 12. P. 12345–12356.
5. Vezakis I. A., Georgas K. et al. EffiSegNet: Gastrointestinal polyp segmentation through a pre-trained EfficientNet-based network with a simplified decoder // 46th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). 2024. P. 1–4.
6. Belozerov I. A., Sudakov V. A. Investigation of machine learning models for medical image segmentation // Preprints of the Keldysh Institute of Applied Mathematics. 2022. P. 37–15.
7. Varshni D., Thakral K. et al. Pneumonia detection using CNN based feature extraction // 2019 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT). 2019. P. 1–7.
8. Rajpurkar P., Irvin J. et al. CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning [Электронный ресурс]: URL:https://arxiv.org/abs/1711.05225 (дата обращения 11.03.25)

9. Khan A. T., Jensen S. M. LEAF-Net: A unified framework for leaf extraction and analysis in multi-crop phenotyping using YOLOv11 // Agriculture. 2025. Vol. 15. No 2. P. 196.

10. Carion N., Massa F. et al. End-to-end object detection with transformers // European conference on computer vision. Cham: Springer International Publishing. 2020. P. 213–229.

11. TBX11K Simplified [Электронный ресурс]: URL:https://www.kaggle.com/datasets/vbookshelf/tbx11k-simplified (дата обращения 12.03.25).

12. NIH Chest X-rays [Электронный ресурс]: URL:https://www.kaggle.com/datasets/nih-chest-xrays/data?select=BBox_List_2017.csv (дата обращения 12.03.25).

13. Nguyen H., Pham H. H. et al. Vinbigdata chest x-ray abnormalities detection [Электронный ресурс]: URL:https://www.kaggle.com/competitions/vinbigdata-chest-xray-abnormalities-detection (дата обращения 12.03.25).

УДК 004.932.72.1

Бабкина А. В., Конев Д. В.

Сегментация почек на ультразвуковых изображениях

Рекомендовано к публикации доцентом Козынченко В. А.

1. Введение. С каждым годом увеличивается количество исследований, использующих нейросетевые технологии в различных областях науки, в том числе и в медицине. С помощью искусственного интеллекта и машинного обучения решаются задачи шумоочистки медицинских изображений, задачи сегментации изображений внутренних органов, детектирования патологий и т. д.

Ультразвуковые исследования (УЗИ) широко используются в медицине как безопасный (из-за отсутствия ионизирующего излучения), неинвазивный и доступный метод исследования внутренних органов. В основе данного метода лежит прием эхо-сигналов ультразвуковых волн, отраженных от внутренних тканей человека. По полученным данным строятся 2D- , 3D- или 4D-изображения.

Сегментация почек на ультразвуковых изображениях может использоваться при диагностике заболеваний. Например, при диагностике мочекаменной болезни подобная система позволила бы быстро и точно сопоставить объемы компьютерной томографии и УЗИ срезов, вместо ручного/полуавтоматического сопоставления, распространенного на данных момент в медицинских учреждениях.

Целью данного исследования является сравнение качества сегментации почки на УЗИ при различной степени очистки изображений от спекл-шума и различном способе получения двумерных срезов.

2. Обзор подходов к получению сегментации. Методы сегментации на ультразвуковых изображениях можно разделить на три основные группы: полученные вручную, полуавтоматические и автоматические.

Бабкина Анна Владимировна – магистрант, Санкт-Петербургский государственный университет; e-mail: st076184@student.spbu.ru, тел.: +7(904)331-72-88

Конев Данил Викторович – аспирант, Санкт-Петербургский политехнический университет Петра Великого; e-mail: konev.d@edu.spbstu.ru, тел.: +7(900)654-50-91

Первая группа методов предполагает участие специалиста, который вручную очерчивает границы почки на изображении. Такой подход отличается высокой точностью при работе с качественными изображениями, но требует значительных временных затрат и опыта оператора.

Полуавтоматические методы облегчают процесс сегментации, комбинируя ручную разметку с алгоритмическими подходами. Например, специалист может задавать начальные контуры или опорные точки, после чего алгоритм уточняет границы органа, используя методы активных контуров, кластеризации или графовых моделей [1, 2].

Среди автоматических методов сегментации существуют методы с применением нейросетевых технологий, которые в свою очередь делятся на 2D-подходы (сегментация проводится на каждом срезе объема независимо от соседних срезов) и 3D-подходы (с учетом взаимного расположения срезов). Далее в работе будет использоваться двумерный подход.

3. Обучающие данные. Основными трудностями при работе с УЗИ изображениями почек являются нечеткие границы, низкий контраст изображений и сложные текстуры. Дополнительной трудностью является отсутствие общедоступных обучающих наборов УЗИ снимков с заданной сегментацией почек.

Для проведения экспериментов были созданы собственные наборы данных УЗИ почек. УЗИ-изображения были получены от добровольцев, а их сбор и разметка проводились под контролем специалистов.

Трехмерное ультразвуковое изображение представляет собой объемный массив данных размером $384 \times 384 \times 384$, сформированный путем интерполяции последовательных 2D-срезов (рис. 1). Эти срезы получены при вращении датчика с помощью специального оборудования, которое позволяет точно контролировать его положение в пространстве и угол наклона.

Ручная разметка трехмерных данных является слишком трудоемким процессом. Поэтому для сегментации был использован инструмент *Grow from Seeds* в программе 3D Slicer [3], основанный на полуавтоматическом алгоритме [2].

Для обучения нейронной сети использовались два способа создания изображений на основе одного и того же 3D-объема ультразву-

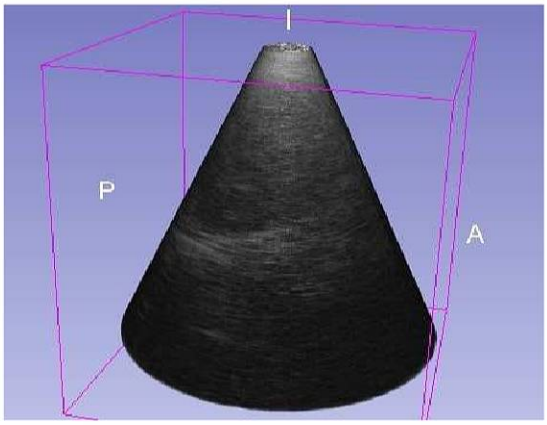


Рис. 1. Трехмерное представление данных

ковых данных с сегментированной почкой.

Первый способ сечения объема данных (рис. 2, слева) заключался в получении изображений путем выполнения разрезов объемов данных в плоскостях Ozx или Ozy. Эти разрезы выполнялись при ортогональном (перпендикулярном) движении среза относительно выбранной оси. При таком подходе из одного 3D-объема данных генерируется 384 изображения. Однако не на каждом из этих изображений присутствует почка, а ее видимая площадь может значительно варьироваться от среза к срезу. Это связано с тем, что почка занимает лишь часть объема, и ее отображение зависит от конкретного положения и ориентации среза.

Второй способ (рис. 2, справа) заключался в создании изображений путем вращения среза вокруг центра 3D-объема под произвольными углами. Если почка расположена в центре объема, то она будет присутствовать на всех полученных изображениях, причем ее площадь на каждом из них будет примерно одинаковой. Это обеспечивает более стабильное и равномерное представление почки в данных, что может быть полезным для обучения модели. При таком подходе из одного 3D-объема данных генерируется $\frac{360}{2\alpha}$ срезов, где α – угол поворота между соседними срезами.

4. Построение нейронной сети. Существует два подхода к сегментации объекта (почки) в трехмерном пространстве: последова-

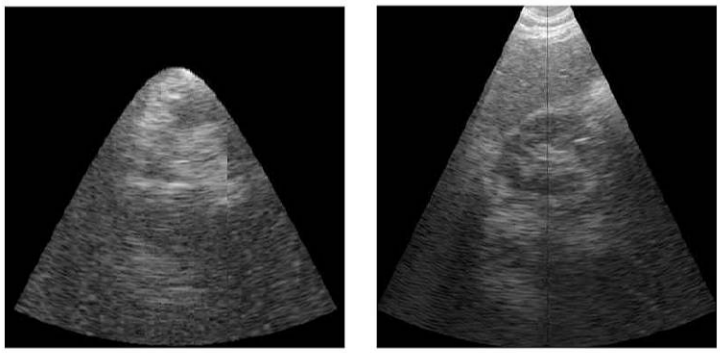


Рис. 2. Пример изображения, полученного поперечными срезами (слева) и срезами вращения (справа)

тельное сегментирование объекта на каждом отдельном срезе и получение трехмерной поверхности при их объединении, либо работа напрямую с трехмерными данными. Для проведения экспериментов был выбран первый подход.

За основу сети взята архитектура UNet [4], состоящая из энкодера и декодера. В качестве энкодера использовалась сеть resnet18. На вход подавались увеличенные до 512×512 изображения срезов, на выходе получались бинарные маски того же размера. В качестве функции активации для всех слоев, кроме последнего, использовалась ReLu, для выходного слоя – функция softmax. В качестве функции потерь использовалась Dice_loss, как распространенная и устойчивая к несбалансированным классам метрика. Всего модель содержала 14,3 млн обучаемых параметров. Итоговая архитектура сети представлена на рис. 3.

5. Сравнение способов сечения обьема данных и влияния шумоочистки входных изображений. Для сравнения качества полученных бинарных масок использовались следующие метрики:

- $\text{Dice_loss} = 1 - \frac{2 \cdot \text{TP}}{2 \cdot \text{TP} + \text{FN} + \text{FP}}$, принадлежащее диапазону $[0, 1]$, где 0 означает полное совпадение предсказанных границ с истинными. Величины TP, FN, FP представляют собой суммы, вычисленные попиксельно;

- Intersection over Union (IoU) – мера, равная отношению пересечения предсказания сети с истинной маской к их объединению:

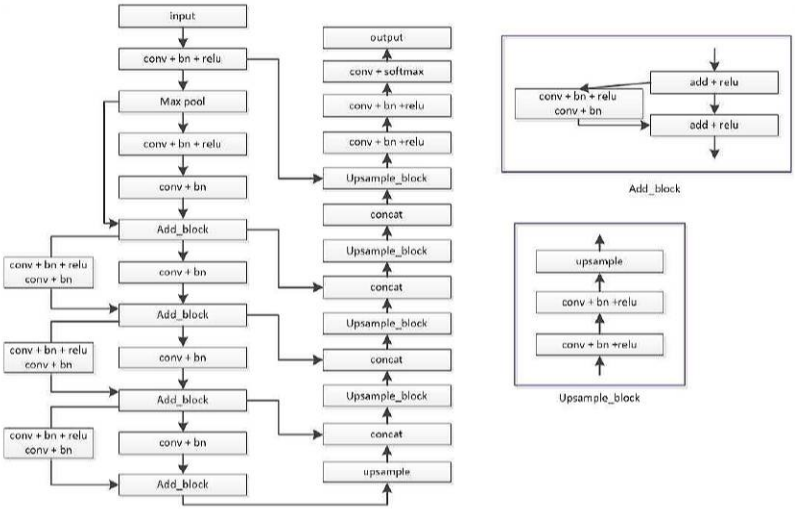


Рис. 3. Архитектура нейронной сети

$\text{IoU}=\frac{\text{TP}}{\text{TP+FN+FP}}$. Значения принадлежат диапазону $[0, 1]$, где 1 означает полное совпадение предсказанных границ с истинными;

- F-score – мера, позволяющая сбалансировать точность и полноту предсказаний: $\text{F}_1 = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$.

Первым экспериментом было проверено влияние степени очистки исходного изображения от спекл-шума на показатели метрик. На вход сети сегментации подавались изображения, полученные первым способом сечения данных. Обучение сети производилось для зашумленных срезов, для очищенных с коэффициентом 0,35 и 1,0 (рис. 4) при помощи сети из [5]. В тренировочную выборку вошли 405 изображений, в валидационную – 130, в тестовую – 160. Результаты экспериментов приведены в таблице 1.

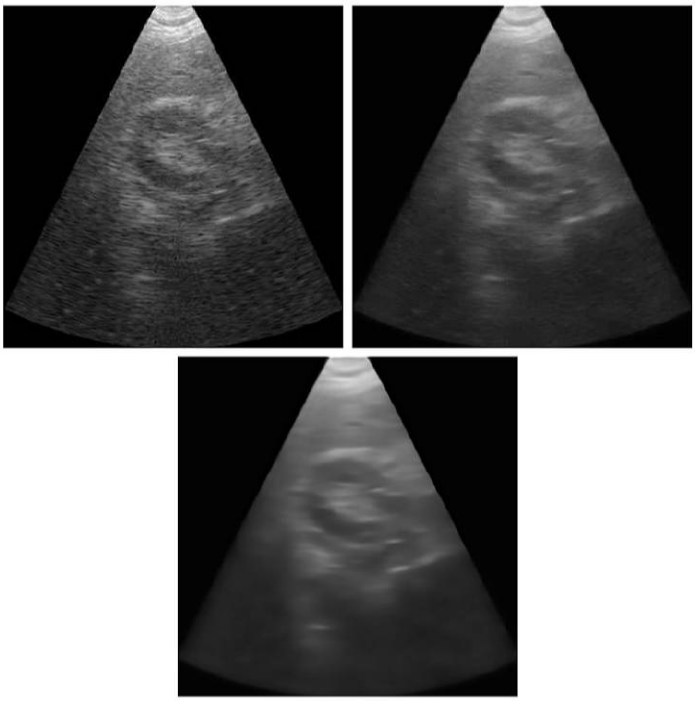


Рис. 4. Пример исходного изображения (вверху слева) и очищенного от спекл-шума с коэффициентом 0,35 (вверху справа) и 1,0 (внизу)

Таблица 1. Влияние степени очистки на сегментацию

	Dice_loss	**F-score**	**IoU**
исходное	0,1096	0,9619	0,9272
шумоочистка 0,35	0,0641	0,9645	0,9319
шумоочистка 1,0	0,0419	0,9641	0,9316

По данным таблицы 1 видно, что наилучшие показатели метрик F-score и IoU дала частичная шумоочистка, а наилучшее значение Dice_loss – полная очистка исходных изображений.

В качестве второго эксперимента было произведено сравнение использования продольных срезов и срезов, полученных за счет вращения. Обучение производилось для 1310 и 1448 изображений соответственно. Соотношение тренировочной, валидационной и тестовой выборок составляло 60%, 20% и 20%. Все изображения были предва-

рительно очищены от спекл-шума (с коэффициентом 0,35). Результаты эксперимента приведены в таблице 2.

Таблица 2. Влияние способа построения срезов на сегментацию

	Dice_loss	**F-score**	**IoU**
поперечные срезы	0,0331	0,9671	0,9368
срезы вращения	0,0147	0,9854	0,9714

Из таблицы 2 видно, что использование срезов вращения лучше, чем использование поперечных срезов. Происходит это благодаря визуальному сходству срезов, возникающему за счет наличия почек во всех кадрах (в одной и той же области кадра) и примерно одинакового занимаемого ей процента площади, что, видимо, облегчает обучение сети. Результат сегментации для одного из срезов вращения приведен на рис. 5.

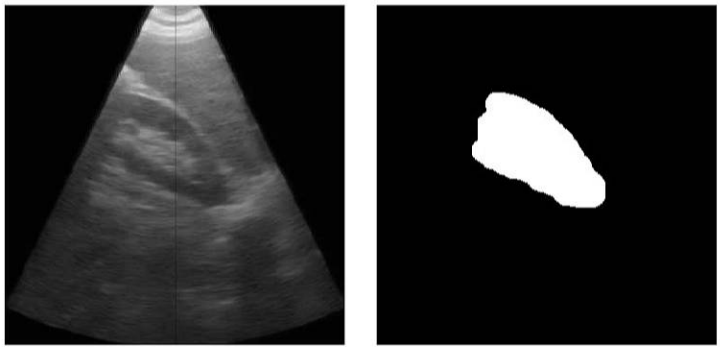


Рис. 5. Срез вращения после очистки от спекл-шума и полученная на выходе сети маска сегментации почки

6. Заключение. В ходе эксперимента было рассмотрено влияние различных степеней шумоочистки и различных способов взятия срезов на качество сегментации почки на ультразвуковых изображениях. В первом эксперименте было выявлено, что полная очистка способствует приросту метрики Dice, а частичная – приросту метрик F-score и IoU. Во втором эксперименте продемонстрирован прирост всех метрик качества при смене способа получения среза с поперечного на вращение.

Детекция почки на УЗИ-снимках является сложной задачей, требующей дальнейших исследований в области предобработки изображений и построения архитектуры нейронной сети.

В дальнейшем полученные результаты планируется использовать для моделирования трехмерной поверхности почки.

Литература

1. Cheng H. D., Shan J. et al. Automated breast cancer detection and classification using ultrasound images: A survey // Pattern recognition. 2010. Vol. 43. No 1. P. 299–317.

2. Zhu L., Kolesov I. et al. An effective interactive medical image segmentation method using fast growcut // MICCAI workshop on interactive medical image computing. 2014.

3. 3D Slicer [Электронный ресурс]: `URL:https://www.slicer.org/` (дата обращения 09.03.25).

4. Ronneberger O., Fischer P., Brox T. U-net: Convolutional networks for biomedical image segmentation // Medical image computing and computer-assisted intervention?MICCAI: 18th international conference. 2015. P. 234–241.

5. Бабкина А. В. Очистка медицинских ультразвуковых изображений от спекл-шума // Процессы управления и устойчивость. 2024. Т. 11. № 1. С. 179–186.

УДК 004.89

Баграмов Н. И.

Автоматическая генерация аннотаций научных публикаций на основе методов обработки естественного языка

Рекомендовано к публикации доцентом Балыкиной Ю. Е.

1. Введение. В последние десятилетия наблюдается значительное увеличение объема научных публикаций. Ежегодно в мире публикуются миллионы научных статей, что приводит к значительным затруднениям в поиске, обработке и анализе релевантной информации. В результате исследователи сталкиваются с рядом проблем:

- существенное усложнение процесса поиска актуальных и значимых работ в своей области знаний;
- увеличение временных затрат на изучение научной литературы, что отрицательно сказывается на скорости научного прогресса;
- повышенная вероятность дублирования исследований, что снижает эффективность использования имеющихся знаний.

В условиях информационного перенасыщения критически важно разработать методы автоматической обработки и сжатия текстов при сохранении их ключевого содержания. Это позволит существенно повысить доступность знаний, оптимизировать работу исследователей и упростить процесс принятия решений в различных областях науки и практики. Таким образом, стоит задача выбора, адаптации и применения методов автоматической суммаризации текстов, обеспечивающих выделение ключевой информации при сохранении ее точности и связности [1].

2. Описание набора данных. Для проведения экспериментов использовался датасет, представленный в работе [2]. Датасет включает 480 пар текстов, содержащих полные версии научных статей и их соответствующие аннотации.

Баграмов Никита Ильич – студент, Санкт-Петербургский государственный университет; e-mail: st097350@student.spbu.ru, тел.: +7(931)335-49-74

В датасет входит следующая информация: полные тексты научных публикаций различных дисциплин, аннотации этих статей.

Статистические характеристики:

- среднее количество предложений в полных текстах варьируется от 95 (медицина) до 241 (история);
- средняя длина текстов от 2028 до 3344 слов в зависимости от дисциплины.

Эти данные позволяют анализировать различные подходы к суммаризации, выявлять закономерности и тестировать модели автоматической генерации аннотаций.

3. Методы суммаризации. Современные методы автоматической генерации аннотаций научных публикаций делятся на экстрактивные, абстрактивные и гибридные. Каждая из этих категорий обладает своими преимуществами и ограничениями, определяющими их эффективность в различных задачах обработки текстовой информации.

3.1. Экстрактивные методы. Экстрактивные методы основываются на выборе наиболее значимых фрагментов исходного текста без изменения их формулировки. Среди наиболее распространенных алгоритмов выделяется PageRank, который представляет текст в виде графа, где вершины соответствуют предложениям, а ребра – степени их семантического сходства. Вес вершины на каждой итерации вычисляется по формуле

$$W(i) = (1 - d) + d \sum_{j \in \mathrm{In}(i)} \frac{W(j)}{O(j)},$$

где $W(i)$ – вес вершины i; d – коэффициент затухания (обычно устанавливается $d = 0{,}85$); знак суммы означает суммирование по вершинам j, которые имеют связь с вершиной i; $O(j)$ – исходящая степень вершины j, т. е. количество дуг, исходящих из нее.

Sentence-BERT (SBERT) представляет собой альтернативный подход, в котором предложения кодируются в векторное пространство, отражающее их семантическую близость. Это позволяет ранжировать текстовые сегменты не только по лексической схожести, но и по смысловой релевантности, что улучшает связность итогового реферата и делает суммаризацию более содержательной.

3.2. Абстрактивные методы. Абстрактивные методы формируют аннотации, используя языковые модели, которые генерируют новые предложения, обобщая и перефразируя исходный текст. Среди наиболее успешных моделей выделяются MBART и ruT5. Модель MBART использует автоэнкодеры для предобучения на многолингвальных данных, а ruT5 является специализированной адаптацией архитектуры T5 для русского языка, работающей в формате «вопрос – ответ» [3].

Главное преимущество абстрактивных методов заключается в их способности не просто извлекать информацию, а формулировать ее в сжатом и осмысленном виде. Однако такие модели зачастую требуют значительных вычислительных ресурсов.

3.3. Гибридные подходы. Для повышения качества суммаризации применяются гибридные методы, комбинирующие экстрактивные и абстрактивные подходы. На первом этапе экстрактивная модель, например SBERT, выделяет наиболее информативные фрагменты текста, а затем абстрактивная модель, такая как ruT5, переформулирует их, создавая осмысленные аннотации. Данный подход позволяет не только выделять ключевые идеи текста, но и переформулировать их в осмысленный и читаемый текст.

4. Результаты. В ходе исследования была проведена серия экспериментов с целью оценки эффективности различных методов суммаризации. Тестирование осуществлялось на корпусе научных публикаций, предварительно обработанных с использованием стандартных процедур токенизации и лемматизации.

Для проведения экспериментов были выбраны четыре модели автоматической суммаризации:

- MBART – многоязычная модель, основанная на архитектуре трансформеров, адаптированная для абстрактивной суммаризации;
- ruT5 – предобученная модель семейства T5, разработанная для русского языка и использующая механизм «text-to-text»;
- TextRank – графовый алгоритм ранжирования предложений на основе семантической близости;
- SBERT+FRED-T5 – гибридный подход, комбинирующий экстрактивную модель Sentence-BERT для выделения ключевых предложений и абстрактивную модель FRED-T5 для их переформулирования.

Реализация моделей осуществлялась на основе доступных библиотек машинного обучения. Для работы с TextRank использовался пакет `networkx` для построения графа связей предложений, а для вычисления семантического сходства применялись эмбеддинги `SentenceTransformer`. В случае SBERT использовалась предобученная модель `paraphrase-multilingual-MiniLM-L12-v2` из библиотеки `sentence-transformers`. Для MBART и ruT5 генерация текстов выполнялась с использованием библиотеки `transformers` от Hugging Face [4].

5. Оценка качества генерации. Для сравнения результатов работы моделей использовались три основных метрики: ROUGE, BLEURT и BERTScore, каждая из которых анализирует текст с разных позиций [5].

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) представляет собой набор метрик, оценивающих точность и полноту совпадений между сгенерированной аннотацией и эталонным текстом. В расчетах использовались показатели ROUGE-1 (совпадение отдельных слов), ROUGE-2 (совпадение биграмм) и ROUGE-L (совпадение длиннейших последовательностей слов). Данная метрика является стандартной для оценки суммаризации, однако она не учитывает смысловую близость текстов. Несмотря на простоту реализации, ROUGE остается важной отправной точкой для оценки качества генерации текстов.

BLEURT (Bilingual Evaluation Understudy with Representations from Transformers) использует предобученную нейросетевую модель, оценивающую сходство между текстами на основе эмбеддингов. Данная метрика позволяет анализировать не только поверхностное совпадение слов, но и более глубокие смысловые связи, что делает ее полезной для оценки абстрактивных методов суммаризации. BLEURT особенно хорошо себя показывает в задачах, где эталонный ответ допускает значительные формулировочные вариации.

BERTScore оценивает тексты, представляя их в виде векторных эмбеддингов, полученных с помощью модели BERT. Метрика рассчитывает косинусное сходство между векторами слов в сгенерированном и эталонном тексте, что позволяет учитывать не только поверхностные лексические совпадения, но и семантическую близость.

Таблица 1. Метрики ROUGE для моделей

Модель	ROUGE-1	ROUGE-2	ROUGE-L
IlyaGusev/mbart	0,059	0,000	0,050
sberbank/ruT5	0,184	0,033	0,110
text_rank	0,122	0,056	0,119
SBERT+FRED-T5	0,203	0,072	0,192

Таблица 2. Метрики BLEURT и BERTScore

Модель	BLEURT	BERTScore
IlyaGusev/mbart	0,182	0,501
sberbank/ruT5	0,322	0,622
text_rank	0,383	0,678
SBERT+FRED-T5	0,452	0,712

Сравнение моделей по данным метрикам (см. таблицу 1, 2) показало, что гибридный метод SBERT+FRED-T5 демонстрирует наилучшие результаты. Экстрактивный алгоритм TextRank обеспечивает приемлемые значения ROUGE, но уступает нейросетевым моделям по когерентности и плавности текста. Абстрактивные модели MBART и ruT5 показали хорошие результаты, но иногда генерировали неосмысленные или слишком общие формулировки.

Пример. Пример аннотации, полученной в результате работы модели SBERT+FRED-T5: «В данной статье рассматривается комбинированный алгоритм сжатия, использующий гибридный алгоритм на первом этапе и zlib на втором этапе. При сжатии изображений, содержащих графики и диаграммы, комбинированный алгоритм демонстрирует на 9,5% более высокую степень сжатия при близком времени выполнения по сравнению с использованием zlib с уровнем 9. В случае изображений, типичных для Windows XP, достигается увеличение степени сжатия на 23,5%. При этом комбинированный алгоритм позволяет увеличить скорость сжатия до 4 раз при сжатии изображений, содержащих текст. В статье представлены результаты тестирования на различных типах изображений, подтверждающие эффективность предложенного подхода. Разработанный алгоритм может быть применен на практике для сжатия кадров экранного видео».

6. Заключение. Проведенные эксперименты подтвердили эффективность автоматической генерации аннотаций, но у применяемых методов есть ограничения: экстрактивные модели не адаптируют текст, а абстрактивные могут допускать смысловые ошибки.

Для улучшения качества суммаризации перспективно использование мультимодальных моделей, учитывающих не только текст, но и графику, а также расширение русскоязычных датасетов.

Автоматическая генерация аннотаций ускоряет поиск и анализ научной литературы, облегчая доступ к ключевой информации. Эти методы также находят применение в реферировании отчетов, анализе новостей и создании конспектов, что делает их универсальным инструментом для обработки текстов.

Литература

1. Белякова А. Ю., Беляков Ю. Д. Обзор задачи автоматической суммаризации текста // Инженерный вестник Дона. 2020. № 2. С. 1–13.

2. Tsanda A., Bruches E. Russian-language multimodal dataset for automatic summarization of scientific papers [Электронный ресурс]: URL: `https://arxiv.org/abs/2405.07886` (дата обращения: 03.03.2025).

3. Gusev I. MBART model for russian summarization [Электронный ресурс]: URL: `https://github.com/IlyaGusev/mbart_ru_sum_gazeta` (дата обращения: 04.03.2025).

4. Береснев В. С. Генерация текстов естественного и программного языка // Процессы управления и устойчивость. 2023. Т. 10. № 1. С. 240–244.

5. Колесов В. В. Алгоритмы для классификации жанров фильма // Процессы управления и устойчивость. 2024. Т. 11. № 1. С. 205–211.

УДК 004.9

Васильев В. А.

Квантизация весов в ключевых компонентах трансформерных архитектур

Рекомендовано к публикации профессором Крылатовым А. Ю.

1. Введение. Трансформер – архитектура глубоких нейронных сетей, представленная в работе [1], стала основой современных моделей LLM, NLP и CV. Однако размеры моделей (от сотни миллионов до миллиардов параметров) затрудняют их использование на мобильных устройствах, особенно в реальном времени. Квантизация весов – преобразование весов из формата с плавающей запятой (float32/float16) в низкочисленный целочисленный (int8/int4) – позволяет решить ряд проблем:

- уменьшить объём памяти. Например, переход float32>int8 уменьшает объём памяти в 4 раза;
- ускорить вычисления за счёт оптимизированных операций с целыми числами, особенно на специализированных аппаратных ускорителях (TPU, NPU, GPU);
- снизить энергопотребление, что критично для ресурсоограниченных устройств (смартфонов и IoT).

2. Методы квантизации. Эти методы можно разделить на три направления в зависимости от этапа, на котором имплеменируется данная технология.

2.1. Посттренировочная квантизация (PTQ). Применяется после обучения модели. Сокращает размер модели (например, в BERT квантизация матриц Q, K, V до int8 сокращает размер модели с 440 МБ до 110 МБ [2, 3]) и состоит из следующих шагов:

1. Калибровка. На небольшом датасете вычисляются диапазоны весов и активаций.

Васильев Владимир Александрович – аспирант, Санкт-Петербургский государственный университет; e-mail: v45il@yandex.ru, тел.: +7(812)363-60-00

2. Масштабирование. Веса преобразуются в целочисленные значения по следующей формуле:

$$W_{\text{int}} = \text{round}\left(\frac{W_{\text{float}}}{S}\right) + Z,$$

где S – масштаб, Z – нулевая точка.

3. Замена операций. Линейные слои заменяются на квантизированные аналоги.

2.2. Квантизация с обучением. Готовить сеть к квантизации можно в процессе обучения, такой подход называется Quantization Aware Training (QAT). Для этого в нейросеть встраивают специальные блоки и в ходе обучения эмулируют квантизованный вывод (инференс). В прямом подходе имитируется округление весов. В обратном подходе используются градиенты исходных весов (Straight-Through Estimator).

Данный вид обучения трудоёмкий и требует больше вычислительных ресурсов, но на выходе получается модель, «приспособленная» к работе с квантизованными значениями и потенциально более точная. Например, для ViT применение QAT снижает потерю точности на ImageNEt с 2,1% (PTQ) до 0,5% [4].

2.3. Гибридные подходы. На практике используют различные комбинации стратегий квантизации в зависимости от струткуры нейросетевой модели:

- смешанная точность (Mixed-precision quantization). Критичные слои сохраняют высокую точность (float16), остальные квантизируются. Подход позволяет добиться баланса между скоростю и качеством модели;
- групповая квантизация. Веса внутри слоя делятся на группы, где для каждой группы подбирается своя шкала квантования с индивидуальными параметрами S и Z. Это может быть эффективно для слоёв с большим разбросом значений.

3. Особенности квантизации в архитектуре трансформер. Трансформеры, включая модели BERT, ViT и GPT, обладают уникальной структурой, что делает их квантование более трудоёмким процессом.

3.1. Слои внимания. Слои внимания (Self-Attention) – основа трансформеров, где матрицы запроса (Q), ключа (K) и значения (V) участвуют в вычислении контекстно зависимых весов. Можно выделить следующие особенности:

- динамический диапазон. Распределение значений в матрицах Q, K, V часто имеют высокую дисперсию. Например, в BERT-base веса матрицы запроса могут варьироваться в диапазоне $[-3{,}8; 4{,}2]$, что усложняет подбор единой шкалы квантования S. Одно из возможных решений – это групповая квантизация [5]. Разделение весов на подгруппы (например, по 64 элемента) с индивидуальными параметрами S и Z. Ещё один вариант – это асимметрическое масштабирование, что подразумевает отдельный расчёт для положительных и отрицательных значений;
- операция Softmax. Эта функция, преобразующая логиты в вероятности, крайне чувствительна к ошибкам квантования. Даже незначительные искажения входных данных (например, из-за int8) могут привести к катастрофическим ошибкам в выходных вероятностях. Решением может выступить сохранение операций Softmax в float16 или калибровка логитов с использованием «логарифмического смещения» для компенсации ошибок [6].

3.2. Feed-forward сети (FFN). Слои в FNN состоят из двух линейных преобразований с активацией (например, GeLU):

- стабильность распределения весов. В отличие от слоёв внимания, веса в FNN обычно имеют меньший динамический диапазон, например, $[-1{,}5; 1{,}5]$ в BERT, что упрощает квантизацию. В GPT-3 квантизация FFN до int8 вызывает падение точности на 0,8% на датасете LAMBADA, тогда как квантование слоёв даёт потерю 2,1% [6];
- активационные функции. Нелинейности (GeLU, ReLU) могут быть квантованы с минимальными потерями, если их входы предварительно отмасштабированы.

3.3. Эмбеддинги. Слой эмбеддингов преобразующий дискретные токены в векторы, может содержать редкие выбросы, что приводит к «длинным хвостам» в распределении. Прямая квантизация может обнулить такие значения. Решением может стать:

- групповая квантизация по измерениям: например, разделение эмбеддингов размерностью 768 на 12 групп по 64 элемента;

- сохранение эмбеддингов в float16 с квантованием только выходных проекций.

3.4. Экспериментальные результаты. Квантизация в сетях с архитектурой трансформер требует детального анализа каждого компонента. Слои внимания и эмбеддинги наиболее чувствительны к потере точности, тогда как FFN и линейные проекции легче оптимизировать. Современные методы, такие как групповая квантизация и QAT, позволяют достичь сжатия в 4–8 раз с сохранением исходной точности больше 98% [2, 4, 6]. В таблице приведены результаты тестирования методов квантизации на разных моделях с архитектурой трансформер.

Таблица. Тестирование методов квантизации [2, 4, 6]

Модель	**Метод**	**Размер (ориг. > квант.)**	**Размер (ориг. > квант.)**
BERT-base	PTQ (int8)	92,3% > 91,5%	440 МБ > 110 МБ
ViT-B/16	QAT (int8)	81,2% > 80,7%	330 МБ > 82 МБ
GPT-3 (175B)	Смешанная	75,1% > 74,3%	3,2 ТБ > 0,8 ТБ

4. Инструменты. Квантизация трансформеров требует специализированных иснтурментов, учитывающих их архитектурные особенности. Ниже приведены ключевые решения.

- TensorFlow Lite/ PyTorch Quantization. Готовые API для методов QAT/PTQ;
- Hugging Face Optimum. Интеграция квантизации моделей типа BERT, GPT-2;
- ONNX Runtime. Поддержка квантизации сетей с архитектурой трансформер для развёртывания. Это open source движок для конвертации моделей из базовых фреймворков в ONNX-формат – высокопроизводительный инструмент для запуска инференсов. В ONNX Runtime (ORT) есть дополнительная оптимизация – квантизация и базовые графовые оптимизации.

5. Заключение. Квантизация весов является ключевой технологией для адаптации современных больших языковых и визуальных моделей к промышленным сценариям применения. Наибольшую эффективность демонстрируют гибридные модели, совмещающие методы QAT и смешанную точность, что обеспечивает сжатие моделей в 4–8 раз при потере точности менее 1–3%.

Можно выделить следующие перспективные направления:

1. Автоматический подбор параметров квантизации. Интеграция с Neural Architecture Search (NAS).
2. Поддержка низкобитных форматов в аппаратных ускорителях.
3. Квантизация для генеративных моделей, таких как GPT-4, Llama, остаётся сложной задачей из-за рекуррентной природы авторегрессии. Решением могут выступить гибридные квантованные KV-кэши.

Литература

1. Vaswani A., Shazeer N., Parmar N. et al. Attention is all you need, Advances in neural information processing systems // NIPS Conference. Vol. 30. 2017.

2. Zafrir O., Boudoukh G., Izsak P. et al. Q8BERT: Quantized 8Bit BERT // NeurIPS. 2019. P. 36–39.

3. Неронов Р. М. Решение задачи разведочного анализа текстов через кластеризацию и аппроксимацию числа кластеров // Процессы управления и устойчивость. 2024. Т. 11. № 1. С. 231–235.

4. Zhihang Yuan, Chenhao Xue, Yiqi Chen, Qiang Wu, Guangyu Sun. PTQ-ViT: post-training quantization for vision transformers with twin uniform quantization [Электронный ресурс] // arXiv. 2023. URL:https://arxiv.org/abs/2111.12293 (дата обращения: 20.02.2025).

5. Bondarenko Y., Nagel M., Blankevoort T. Understanding and overcoming the challenges of efficient transformer quantization // In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. 2021. P. 7947–7969.

6. Yao Z., Yazdani Aminabadi R., Zhang M. et al. ZeroQuant: efficient and affordable post-training quantization for large-scale transformers [Электронный ресурс] // NeurIPS. 2022. URL:https://arxiv.org/abs/2206.01861 (дата обращения: 28.02.2025).

УДК 519.673

Вашукова А. М.

Применение нейронных обыкновенных дифференциальных уравнений для моделирования динамических систем

Рекомендовано к публикации доцентом Головкиной А. Г.

1. Введение. Научная область нейронных обыкновенных дифференциальных уравнений (Neural ODEs) представляет собой одно из самых перспективных направлений на стыке математики и машинного обучения. Этот подход объединяет традиционные численные методы решения обыкновенных дифференциальных уравнений (ОДУ) с гибкостью нейронных сетей, что позволяет эффективно моделировать динамические системы и предсказывать их эволюцию на основе данных [1].

2. Методы. В данной работе реализованы и протестированы пять методов на основе Neural ODE. Все методы используют общую архитектуру и параметры обучения (Архитектура: RNN (GRU) с последующим решением ODE. Количество эпох: 50. Функция потерь: MSE (Mean Squared Error). Оптимизатор: Adam. Скорость обучения: 0,01). Для решения ОДУ используется метод dopri5 (метод Рунге – Кутты пятого порядка, предложенный Дормандом и Принсом [2, 3]) из библиотеки torchdiffeq. Для проведения экспериментов использовались данные о влажности в городе Дели, Индия, а также синтетические данные (синусоида с шумом).

2.1. Neural ODE. Метод Neural ODE заменяет дискретные слои нейронных сетей на непрерывные динамические системы, описываемые ОДУ [1]. Основное уравнение

$$\frac{dh(t)}{dt} = f(h(t), t, \theta), \tag{1}$$

где $h(t)$ – состояние системы во времени t, а f – параметризованная функция (обычно нейронная сеть). Уравнение (1) является основой

Вашукова Анна Михайловна – студент, Санкт-Петербургский государственный университет; e-mail: st097357@student.spbu.ru, тел.: +7(960)015-01-11

для моделирования динамических систем, где изменение состояния системы во времени определяется функцией f.

Решение ОДУ может быть записано в интегральной форме

$$h(t_1) = h(t_0) + \int_{t_0}^{t_1} f(h(t), t, \theta)\, dt,$$

или найдено с помощью численного метода, который обозначим

$$h(t_1) = \text{ODESolve}(h(t_0), f, t_0, t_1, \theta).$$

Для численного решения ОДУ в численном эксперименте используется метод dopri5. Этот метод обеспечивает высокую точность и эффективность при решении нежестких систем дифференциальных уравнений.

Для более простых задач может использоваться метод Эйлера, который является одним из самых простых численных методов решения ОДУ. Однако этот метод менее точен, чем dopri5.

Методы Neural ODE особенно полезны для задач, где требуется моделирование непрерывных процессов. Они также эффективны для задач с длинными временными рядами, так как не требуют фиксированного числа слоев, как в традиционных RNN.

2.2. p-ODE-RNN. В этом методе на основе вероятности $p = 1/t$ используется либо RNN, либо ODE-RNN для обработки временного ряда. Это позволяет гибко комбинировать преимущества дискретных и непрерывных моделей [4]. Основное уравнение для RNN

$$h_t = \sigma(W_h h_{t-1} + W_x x_t + b),$$

где h_t – скрытое состояние на шаге t, x_t – входные данные, W_h, W_x – весовые матрицы, b – смещение, а σ – функция активации.

Основные уравнения для ODE-RNN

$$h'_t = \text{ODESolver}(h_{t-1}, f_\theta, t),$$

$$h_t = \text{RNNCell}(h'_t, x_t),$$

где f – параметризованная функция (обычно нейронная сеть).

Выбор между RNN и ODE-RNN осуществляется на основе вероятности $p = 1/t$, что позволяет адаптировать модель к данным с переменной структурой.

2.3. Augmented Neural ODE (ANODE). Данный метод расширяет пространство признаков, добавляя дополнительные скрытые состояния для улучшения выразительности модели [5]. Основное уравнение

$$\frac{d}{dt}\begin{bmatrix} h(t) \\ h(t) \end{bmatrix} = \begin{bmatrix} f(h(t), t, \theta) \\ g(h(t), t, \phi) \end{bmatrix},$$

где $h(t)$ – дополненное состояние. Это позволяет моделировать более сложные динамические системы, где стандартные Neural ODE могут быть недостаточно выразительными. Дополнительные состояния $h(t)$ могут быть использованы для кодирования дополнительной информации о системе, что улучшает качество предсказаний.

2.4. Hamiltonian Neural Networks (HNN). Данный метод используют гамильтонову механику для моделирования физических систем, в которой величины q и p представляют собой обобщенные координаты и импульсы, соответственно. Эти переменные используются для описания состояния системы и ее динамики [6]. Основное уравнение

$$\dot{q}(t) = \frac{\partial H}{\partial p}, \quad \dot{p}(t) = -\frac{\partial H}{\partial q},$$

где $H(q, p, t)$ – гамильтониан системы, который представляет собой функцию, описывающую полную энергию системы. В HNN гамильтониан параметризуется нейронной сетью

$$H(q, p) = \mathrm{NN}(q, p; \theta),$$

где NN – нейронная сеть с параметрами θ. Это позволяет моделировать системы, где важно сохранять физические законы, такие как сохранение энергии.

2.5. Neural Controlled Differential Equations (NCDE). Данный метод добавляет управляющее воздействие для моделирования систем, зависящих от внешних факторов [7]. Основное уравнение

$$\frac{dh(t)}{dt} = f(h(t), u(t), t, \theta),$$

где $u(t)$ – управляющее воздействие. Это позволяет моделировать системы, где состояние зависит не только от текущего состояния, но и от внешних входных данных. Управляющее воздействие $u(t)$ может быть представлено как функция времени или как выход другой нейронной сети.

3. Результаты численного эксперимента. В таблице представлены результаты экспериментов на синтетических данных (синусоида с шумом) и реальных данных (влажность в г. Дели). Для каждого метода указаны время обучения и значение функции потерь (MSE).

Таблица. Результаты экспериментов

	Синтетические данные		**Реальные данные**	
Метод	**Время, с**	**MSE**	**Время, с**	**MSE**
Neural ODE	12,21	0,0127	31,05	0,0157
p-ODE-RNN	10,27	0,0142	30,01	0,0126
Augmented ODE	9,58	0,0123	40,18	0,0089
HNN	7,05	0,0135	33,97	0,0104
NCDE	11,94	0,0193	41,88	0,0150

4. Заключение. В работе реализованы и протестированы пять методов на основе Neural ODE. Наилучшие результаты по точности (MSE) на реальных и синтетических данных показал метод Augmented Neural ODE [5], так как дополнительные состояния позволяют моделировать системы с более высокой размерностью и сложностью. Данный метод может кодировать больше информации о системе, что приводит к более точным предсказаниям. По скорости обучения на синтетических данных лучшим оказался Hamiltonian Neural Network [6]. Это связано с тем, что параметризация гамильтониана нейронной сетью проще, чем в других методах, таких как Augmented Neural ODE, где требуется моделировать дополнительные состояния. Меньшая сложность модели приводит к более быстрому обучению. На реальных данных быстрее всего оказался p-ODE-RNN [4]. На основе вероятности $p = 1/t$ модель выбирает, какой из методов (RNN или ODE-RNN) использовать на каждом шаге. Это позволяет избежать избыточных вычислений, которые могут возникать при использовании только Neural ODE [1]. Остальные методы также показали хорошие результаты по скорости и MSE.

Литература

1. Chen R. T. Q., Rubanova Y., Bettencourt J., Duvenaud D. Neural ordinary differential equations // Advances in Neural Information Processing Systems. 2018. No 31. P. 6571–6583.

2. Dormand J. R., Prince P. J. A family of embedded Runge–Kutta formulae // Journal of Computational and Applied Mathematics. 1980. No 6. P. 19–26.

3. Олемской И. В., Еремин А. С., Матросов А. В. Прямой метод интегрирования систем обыкновенных дифференциальных уравнений второго порядка // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2024. Т. 20. № 3. С. 324–334.

4. Маргасов А. О. О нейронных обыкновенных дифференциальных уравнениях и их вероятностном расширении // Известия Коми научного центра УрО РАН. 2021. № 6(52). С. 14–19.

5. Dupont E., Doucet A., Teh Y. W. Augmented neural ODEs // Advances in Neural Information Processing Systems. 2019. No 32. P. 3138–3149.

6. Greydanus S., Dzamba M., Yosinski J. Hamiltonian neural networks // Advances in Neural Information Processing Systems. 2019. No 32. P. 15379–15389.

7. Kidger P., Morrill J., Foster J., Lyons T. Neural controlled differential equations for irregular time series // Advances in Neural Information Processing Systems. 2020. No 33. P. 6696–6707.

УДК 004.82

Гаврилов Н. О., Корхов В. В., Пен Е. А., Токарев А. Э.

Разработка алгоритма построения графа знаний на основе патентных данных

1. Введение. В данной статье предлагается способ представления патентных данных в виде графа для выявления неявных взаимосвязей. Для этого векторизуются тексты патентов, что позволяет находить контекстно схожие документы и строить граф. В работе описан процесс построения графа, рассматривается алгоритм кластеризации графов для выявления структурных паттернов в данных.

Графы знаний представляют собой структуру данных, которая отображает отношения между различными сущностями и используется для решения широкого спектра задач, таких как поиск информации [1], рекомендательные системы [2] и обработка естественного языка. В отличие от традиционных графов знаний, в которых узлы содержат конкретные сущности и отношения между ними выражаются через триплеты вида ⟨сущность – отношение – сущность⟩ [3], в данном подходе узлы графа содержат текстовые данные, а связи между ними выявляются с помощью векторных представлений этих текстов.

Основная сложность построения графов знаний из патентных данных заключается в сложности текста, который часто содержит специализированную терминологию и юридические формулировки. Для эффективной обработки таких данных необходимы алгоритмы, учитывающие особенности русского языка, поскольку методы, разработанные для английского, не всегда применимы [4].

Гаврилов Никита Олегович – студент, Санкт-Петербургский государственный университет; e-mail: st102441@student.spbu.ru, тел.: +7(983)250-87-94

Корхов Владимир Владиславович – доцент, Санкт-Петербургский государственный университет; e-mail: v.korkhov@spbu.ru, тел.: +7(812)428-47-83

Пен Евгений Александрович – исполнительный директор ООО «БиоГеоХаб»; e-mail: evgeniypen@gmail.com, тел.: +7(812)428-47-83

Токарев Алексей Эдуардович – руководитель проектов АО «Единый депозитарий результатов интеллектуальной деятельности»; e-mail: Atokarev.e@gmail.com, тел.: +7(812)428-47-83

2. Постановка задачи. Цель данной работы – предложить способ выявления связей патентных данных. Для достижения поставленной цели определены следующие задачи: а) разработка и реализация алгоритма построения графа; б) применение алгоритма кластеризации на графах; в) интерпретация результатов.

3. Представленное решение. На рис. 1 представлен процесс преобразования исходного набора патентных данных в граф посредством последовательного выполнения четырех этапов.

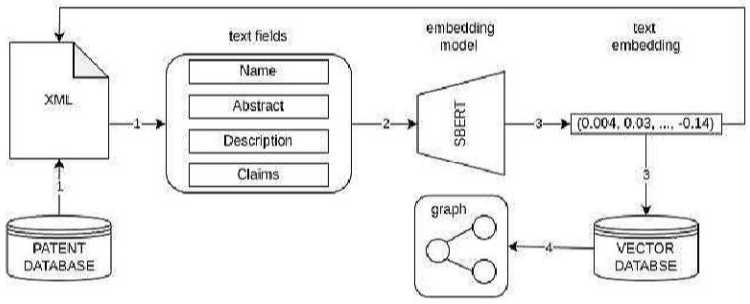


Рис. 1. Алгоритм построения графа

1. Извлекаются основные текстовые поля каждого патента, которые хранятся в виде XML-файлов. Для дальнейшей векторизации будем использовать поле формула изобретения (Claims), так как считаем, что оно является наиболее информативным, передает суть патента.
2. Каждый из выделенных текстовых блоков проходит процедуру векторизации. Для этого используется модель на архитектуре трансформер, поддерживающая русский язык как основной или как один из набора возможных. Была выбрана модель ruRoBERTa-large, поскольку она показывает высокие результаты в задачах Russian SuperGLUE [5].
3. Используем векторную базу данных для эффективного хранения и поиска документов. В нашей реализации используется библиотека FAISS, поддерживающая работу с GPU.
4. Каждому патенту сопоставляется узел графа, а ребра между узлами отражают семантическую близость документов. Семан-

тически схожие патенты определяются на основе фиксированного количества ближайших соседей или порогового значения косинусного расстояния.

Построенный граф патентов служит основой для дальнейшего анализа. Для выявления структурных закономерностей и группировки схожих документов применяется алгоритм кластеризации Leiden [6], обеспечивающий высокую внутреннюю связность узлов. Полученные кластеры суммаризируются с помощью больших языковых моделей, которые автоматически извлекают значимые термины и формируют содержательные описания. На их основе формируется новый граф, в котором каждый узел представляет собой отдельный кластер. Следовательно, уменьшается изначальный размер графа.

4. Анализ результатов. Для исследования использовался корпус из 329826 патентов, представленных в формате XML, из базы данных RUPAT, предоставленной Федеральным институтом промышленной собственности (ФИПС). База данных RUPAT содержит полные тексты патентов России, включая описание и формулы изобретений, рефераты, а также данные о заявителе и авторе. Текстовые поля патентных документов были извлечены и сохранены в реляционной СУБД PostgreSQL. Первый граф, построенный по алгоритму из пункта 3, содержит 329826 узлов и 2403178 ребер. В данной реализации алгоритма для каждого патента было найдено 9 ближайших соседей.

Важную роль играет равномерность разбиения кластеров, поскольку чрезмерно крупные или разреженные кластеры затрудняют дальнейший анализ и снижают информативность. На рис. 2 представлены результаты изменения метрики modularity, нормализованной энтропии Шеннона и количества кластеров от параметра resolution, который позволяет контролировать гранулярность кластеризации. Метрика modularity рассчитывается по формуле

$$Q = \frac{1}{2m} \sum_{ij} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta(c_i, c_j),$$

где m – общее количество ребер в графе; A_{ij} – элемент матрицы смежности, равный 1, если между узлами i и j есть ребро, и 0 – в противном случае; k_i и k_j – степени узлов i и j (количество ребер, инцидентных узлам); $\delta(c_i, c_j)$ – функция Кронекера.

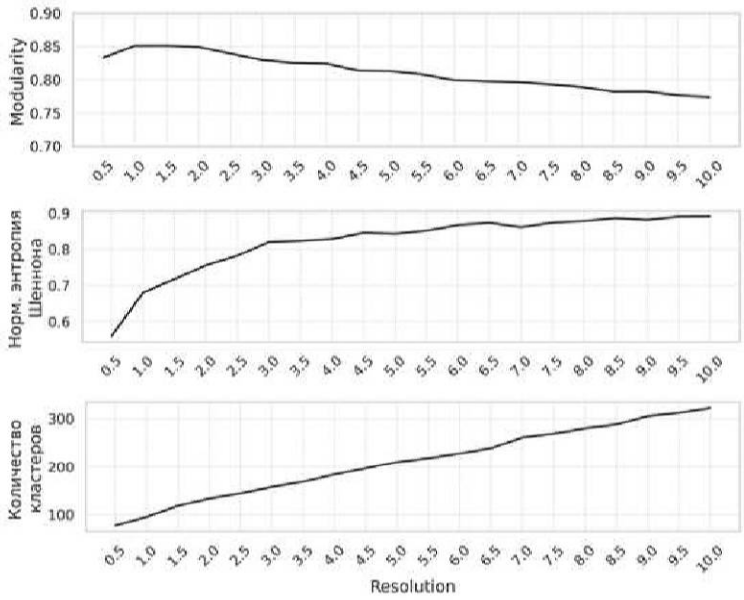


Рис. 2. Зависимость а) modularity, б) энтропии Шеннона, в) количества кластеров от параметра resolution

На рис. 3 показаны изменения тех же показателей только для указанного ограничения в количестве узлов в каждом кластере: реализация алгоритма в библиотеке leidenalg [7] позволяет указать параметр max_comm_size (Maximum community size), который ограничивает размер кластера.

Алгоритмы оптимизации модульности, такие как Leiden, не всегда справляются с обнаружением маленьких кластеров, если их размер меньше определенного порога [8]. Поэтому метрика modularity не всегда является информативной при определении качества разбиения.

На рис. 2, 3 видно, что с увеличением числа кластеров растет равномерность их распределения, но одновременно уменьшается значение метрики modularity, что, как показано ранее, не является критичным. Установим, что значение нормализованной энтропии Шен-

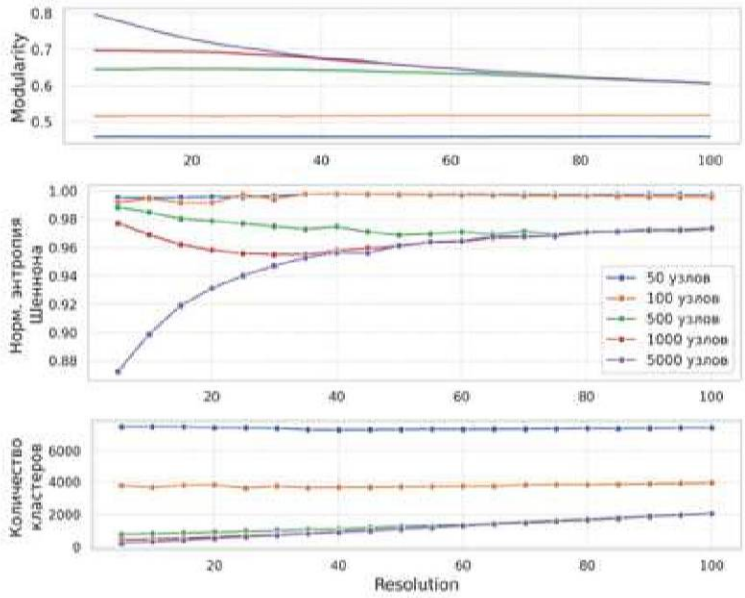


Рис. 3. Зависимость а) modularity, б) энтропии Шеннона, в) количества кластеров от resolution при различных max_comm_size

нона 0,9 является приемлемым, и на этой основе устанавливаем параметр resolution, равным 10, что обеспечивает более равномерное распределение узлов по кластерам. В результате получаем 322 кластера с максимальным количеством узлов в кластере, равным 4529, нормализованной энтропией Шеннона 0,89 и значением modularity, равным 0,77.

Таким образом, новый граф содержит 322 узла и 7370 ребер: два кластера имеют связь, если существует хотя бы одно ребро в исходном графе, которое соединяет узлы из этих кластеров. На рис. 4 представлен подграф, где размер узла отображает количество патентов в кластере. Вес каждого ребра в новом графе определяется количеством ребер в исходном графе, которые связывают узлы двух соответствующих кластеров. Использование большой языковой модели, в нашем случае Mistral 7B, дообученной на русском корпусе

текстов [9], позволяет выделить ключевые слова, описывающие кластеры.

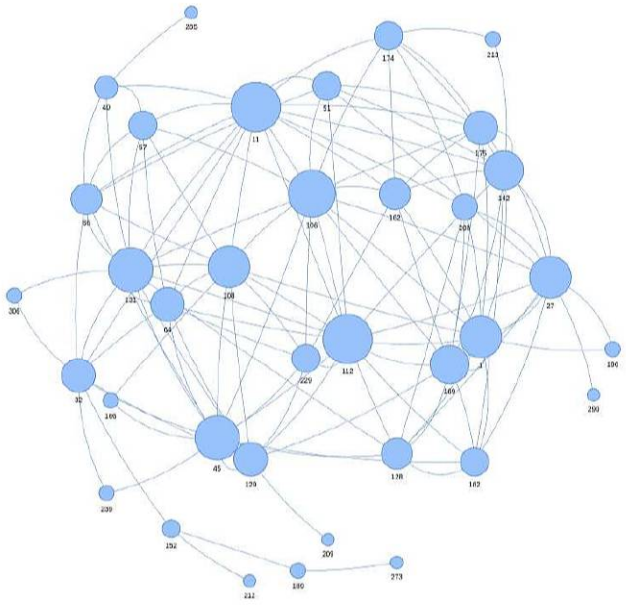


Рис. 4. Подграф конечного графа

5. Заключение. В данной работе предложен алгоритм построения графа знаний на основе патентных данных, который позволяет выявлять семантические связи между патентами. Применение векторизации текста и кластеризации графов с использованием алгоритма Leiden способствует выявлению скрытых структурных закономерностей и упрощению анализа больших объемов данных. Результаты показывают, что предложенный подход улучшает интерпретацию патентных данных, обеспечивая более компактное и информативное представление. Этот метод имеет значительный потенциал для дальнейших исследований и практического применения в области интеллектуальной собственности и анализа научных достижений.

Литература

1. Wang S., Fan W., Feng Y., Ma X., Wang S., Yin D. Knowledge graph retrieval-augmented generation for LLM-based recommendation [Электронный ресурс]: URL:https://arxiv.org/pdf/2501.02226 (дата обращения: 28.09.24).
2. Yang Y., Huang C., Xia L., Huang C. Knowledge graph self-supervised rationalization for recommendation [Электронный ресурс]: URL:https://arxiv.org/pdf/2307.02759 (дата обращения: 19.10.24).
3. Bruches E., Pauls A., Batura T., Isachenko V. Entity recognition and relation extraction from scientific and technical texts in russian [Электронный ресурс]: URL:https://arxiv.org/pdf/2011.09817 (дата обращения: 03.11.24).
4. Siddharth L., Blessing L., Wood K., Luo J. Engineering knowledge graph from patent database [Электронный ресурс]: URL:https://arxiv.org/pdf/2106.06739 (дата обращения: 05.11.24).
5. Zmitrovich D., Abramov A., Kalmykov A., Tikhonova M., Taktasheva E. et al. A family of pretrained transformer language models for russian [Электронный ресурс]: URL:https://arxiv.org/pdf/2309.10931 (дата обращения: 07.12.24).
6. Traag V., Waltman L., van Eck N. From Louvain to Leiden: guaranteeing well-connected communities [Электронный ресурс]: URL:https://arxiv.org/pdf/1810.08473 (дата обращения: 20.01.25).
7. Leidenalg [Электронный ресурс]: URL:https://leidenalg.readthedocs.io/en/stable/index.html (дата обращения: 15.02.25).
8. Fortunato S., Barthelemy M. Resolution limit in community detection [Электронный ресурс]: URL:https://arxiv.org/pdf/physics/0607100 (дата обращения: 22.01.25).
9. Russian Mistral-based chatbot [Электронный ресурс]: URL:https://huggingface.co/IlyaGusev/saiga_mistral_7b_lora (дата обращения: 17.02.25).

УДК 519.233.33

Голофастов Л. Д., Миронов А. В.

Реализация и анализ мощности критериев согласия с экспоненциальным распределением

Рекомендовано к публикации доцентом Гориховским В. И.

1. Введение. Критерии согласия – это статистические методы, позволяющие проверить гипотезу о соответствии эмпирических данных заданному теоретическому распределению. Они играют ключевую роль в различных областях науки и техники, таких как инженерия, экономика, медицина и экология [1–3].

Проверка гипотез о распределении критична для выбора адекватных моделей в прикладных исследованиях. Ошибочное принятие неверного распределения может привести к некорректным выводам, например, в задачах прогнозирования или оптимизации.

Экспоненциальное распределение, характеризующееся свойством отсутствия памяти, широко используется в инженерии, биологии и телекоммуникациях. Однако его проверка осложняется тем, что параметр масштаба часто неизвестен и оценивается по выборке.

В данной работе проводится сравнительный анализ мощности критериев согласия с экспоненциальным распределением с использованием фреймворка PySATL [4].

Несмотря на множество существующих критериев, их мощность существенно зависит от типа альтернативного распределения и размера выборки. Новые инструменты, такие как PySATL, позволяют автоматизировать расчеты и уточнить рекомендации по выбору критериев в конкретных сценариях.

В данной работе представлена реализация критериев согласия с экспоненциальным распределением, проведенный эксперимент по сравнению мощности критериев, а также анализ полученных результатов.

Голофастов Лев Дмитриевич – студент, Санкт-Петербургский государственный университет; e-mail: l.golofastov@mail.ru, тел.: +7(919)554-53-45

Миронов Алексей Владиславович – аспирант, Санкт-Петербургский государственный университет; e-mail: alexmir2015@yandex.ru, тел.: +7(911)146-14-40

2. Разработка критериев. Была создана методология и программная платформа, обеспечивающие эффективную оценку мощности статистических критериев согласия. Данный инструментарий позволяет исследователям выбирать оптимальные методы проверки гипотез с учетом объема данных и характера предполагаемых отклонений от нулевого распределения. В основе разработки лежат современные технологии, включая язык Python и библиотеки NumPy и SciPy, что гарантирует простоту интеграции, адаптивность и воспроизводимость анализа.

В рамках проекта в открытом репозитории фреймворка PySATL были реализованы специализированные критерии согласия для проверки гипотез о соответствии данных экспоненциальному распределению, которые можно классифицировать по источнику информации для принятия решения об отклонении нулевой гипотезы:

Epps – Pulley (EP) [5], Kolmogorov – Smirnov (KS) [6], Atkinson (ATK) [7], Cox – Oakes (CO) [8], Cramer – von Mises (CVM) [9], Deshpande (DSP) [10], Epstein (EPS) [11], Frozini (FZ) [12], Gini (Gini) [13], Pietra (PT) [13], Gnedenko (GD) [14], Harris (HM) [15], Hegazy – Green 1 (HG1) [16], Hegazy – Green 2 (HG2) [16], Kimber – Michael (KM) [17], Kochar (KC) [18], Lorenz (LZ) [19], Moran (MN) [20], Shapiro – Wilk (SW) [21], WE test [22], Wong – Wong [23].

3. Эксперимент по сравнению мощности критериев. В данном разделе рассмотрен проведенный эксперимент по расчету мощностей критериев экспоненциальности средствами фреймворка PySATL. Также проанализированы полученные результаты и выделены лучшие критерии.

3.1. Организация эксперимента. Эксперимент состоит из следующих этапов:

1. *Выбор альтернативных распределений.* Рассматриваются две группы распределений: симметричные и асимметричные.
2. *Генерация выборок из альтернативных распределений.* Генерируются по 1000 выборок из альтернативных распределений размерами 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000.
3. *Расчет предельных распределений статистик критериев.* Для статистики каждого критерия было рассчитано ее предельное распределение методом Монте-Карло. Для нулевой гипотезы (экспоненциального распределения с параметром $\lambda = 1$) было

сгенерировано 100000 выборок упомянутых выше размеров и рассчитана статистика на каждой из них.

4. *Расчет критических значений.* Были получены критические значения из предельных распределений статистик для уровня значимости 0,05.
5. *Расчет мощности.* Расчет мощности производился для каждой альтернативы.
6. *Анализ результатов.* Полученные данные были проанализированы и сформулированы рекомендации по использованию критериев.

Симметричные распределения, рассмотренные в рамках данного эксперимента:

- три случая распределения Бета – $Beta(a; b)$: $Beta(0{,}5; 0{,}5)$, $Beta(1; 1)$ и $Beta(2; 2)$, где a и b – параметры формы;
- три случая распределения Коши – $Cauchy(t; s)$: $Cauchy(0; 0{,}5)$, $Cauchy(0; 1)$ и $Cauchy(0; 2)$, где t и s – параметры положения и масштаба;
- один случай распределения Лапласа – $Laplace(t; s)$: $Laplace(0; 1)$, где t и s – параметры положения и масштаба;
- один случай логистического распределения – $Logistic(t; s)$: $Logistic(2; 2)$, где t и s – параметры положения и масштаба;
- четыре случая распределения Стьюдента – $Student(v)$: $Student(1)$, $Student(2)$, $Student(4)$ и $Student(10)$, где v – число степеней свободы;
- пять случаев распределения Тьюки – $Tukey(A)$: $Tukey(0.14)$, $Tukey(0{,}5)$, $Tukey(2)$, $Tukey(5)$ и $Tukey(10)$, где A – параметр формы;
- пять случаев нормального распределения – $N(m; v)$: $N(0; 1)$, $N(0; 4)$, $N(4; 1)$, $N(-4; 1)$, $N(0; 0{,}5)$, где m и v – параметры положения и масштаба.

Асимметричные распределения, рассмотренные в рамках данного эксперимента:

- четыре случая распределения Бета – $Beta(a; b)$: $Beta(2; 1)$, $Beta(2; 5)$, $Beta(4; 0{,}5)$ и $Beta(5; 1)$;
- три случая распределения Хи-квадрат – $\chi^2(v)$: $\chi^2(1)$, $\chi^2(4)$ и $\chi^2(10)$, где v – число степеней свободы;

- шесть случаев распределения Гамма – $Gamma(a; b)$: $Gamma(2; 2)$, $Gamma(3; 2)$, $Gamma(5; 1)$, $Gamma(9; 1)$, $Gamma(15; 1)$ и $Gamma(100; 1)$, где a и b – параметры формы и масштаба;
- один случай распределения Гумбеля – $Gumbel(t; s)$: $Gumbel(1; 2)$, где t и s – параметры положения и масштаба;
- один случай логнормального распределения – $Lognormal(t; s)$: $LN(0; 1)$, где t и s – параметры положения и масштаба;
- четыре случая распределения Вейбулла – $Weibull(a; b)$: $Weibull(0{,}5; 1)$, $Weibull(1; 2)$, $Weibull(2; 3{,}4)$ и $Weibull(3; 4)$, где a и b – параметры формы и масштаба;
- три случая экспоненциального распределения – $Exp(l)$: $Exp(1)$, $Exp(4)$, и $Exp(10)$, где l – это обратный коэффициент масштаба;
- пять случаев распределения Гомпертца – $Gompertz(e; b)$: $Gompertz(0{,}1; 1)$, $Gompertz(2; 1)$, $Gompertz(3; 1)$, $Gompertz(1; 2)$, $Gompertz(1; 3)$, где e и b – параметры формы и масштаба;

3.2. Результаты и анализ. Для каждой пары критерия и альтернативы определялась статистическая мощность. На основе результатов эксперимента были посчитаны средние мощности критериев на объемах выборки 10, 100, 1000, отсортированы в порядке невозрастания и выделены 10 лучших критериев. Наиболее точные критерии против симметричных и асимметричных альтернативных распределений представлены в таблицах 1, 2. Также в таблице 3 представлена информация о наиболее точных критериях против всех альтернатив.

Таблица 1. Топ критериев против симметричных альтернатив

№ №	10	100	1000
1	FZ (0,921)	CVM (1,000)	CVM (1,000)
2	CVM (0,916)	HG1 (1,000)	HG1 (1,000)
3	KS (0,905)	HG2 (1,000)	HG2 (1,000)
4	HG1 (0,766)	FZ (1,000)	FZ (1,000)
5	EP (0,757)	KS (1,000)	KS (1,000)
6	WE (0,731)	GD (0,992)	GD (1,000)
7	SW (0,599)	SW (0,982)	SW (0,994)
8	GD (0,582)	HM (0,853)	HM (0,871)
9	KC (0,544)	EP (0,773)	EP (0,773)
10	LZ (0,529)	WE (0,773)	WE (0,773)

Таблица 2. Топ критериев против асимметричных альтернатив

№ №	10	100	1000
1	FZ (0,543)	HG1 (0,926)	HG1 (0,964)
2	CVM (0,540)	HG2 (0,862)	HG2 (0,963)
3	CO (0,531)	FZ (0,834)	CVM (0,890)
4	MN (0,525)	CVM (0,830)	KS (0,890)
5	KC (0,520)	KS (0,811)	FZ (0,890)
6	LZ (0,519)	KM (0,792)	KM (0,851)
7	KS (0,512)	ATK (0,789)	ATK (0,847)
8	ATK (0,507)	SW (0,753)	EPS (0,787)
9	KM (0,495)	GD (0,737)	SW (0,774)
10	GD (0,474)	KC (0,719)	GD (0,774)

Таблица 3. Топ критериев против всех альтернатив

№ №	10	100	1000
1	FZ (0,716)	HG1 (0,960)	HG1 (0,980)
2	CVM (0,712)	HG2 (0,925)	HG2 (0,980)
3	KS (0,692)	FZ (0,910)	CVM (0,941)
4	HG1 (0,577)	CVM (0,908)	KS (0,941)
5	KC (0,531)	KS (0,897)	FZ (0,940)
6	GD (0,524)	SW (0,858)	GD (0,878)
7	LZ (0,523)	GD (0,854)	SW (0,875)
8	KM (0,502)	HM (0,722)	HM (0,761)
9	SW (0,454)	KM (0,699)	KM (0,729)
10	HG2 (0,418)	KC (0,660)	LZ (0,672)

Ключевые наблюдения:

1. Мощность критериев с ростом объема выборки в среднем возрастает.
2. Уже на объеме выборки 100 критерии хорошо распознают альтернативу.
3. Во всех сводных таблицах первые пять лучших критериев повторяются. Более того, у каждого из них источником информации для принятия решения об отклонении нулевой гипотезы является эмпирическая функция распределения, в то время как остальные критерии имеют другие источники информации.
4. Против асимметричных распределений ни один критерий не смог в 100% случаев отличить альтернативу. Это связано с тем, что в эксперимент было включено экспоненциальное распределение. Однако критерии HG1 и HG2 способны с какого-то момента отвергать экспоненциальное распределение, если параметр сильно отличается от нулевой гипотезы.

5. Несмотря на то, что в список лучших критериев попали критерии с линейной ($O(n)$) вычислительной сложностью расчета статистики, лучшими остаются критерии со сложностью вычисления статистики $O(n \log n)$.
6. В целом критерий HG1 показал лучшие результаты.

4. Заключение. В данной работе реализованы 21 критерий согласия с экспоненциальным распределением и интегрированы в проект PySATL [24]. Также был проведен сравнительный анализ критериев по мощности и выделены наиболее точные из них (CVM, FZ, HG1, HG2, KS). Однако, будучи самыми точными даже на малых выборках, они остаются вычислительно сложными. Проведенный анализ позволил систематизировать подходы к проверке гипотез и выявить ключевые закономерности в их эффективности. Реализация критериев в PySATL упростила процесс проверки гипотез за счет автоматизации генерации критических значений методом Монте-Карло и расчета мощности. Это снижает риск ошибок, связанных с ручными вычислениями, и ускоряет анализ данных в прикладных исследованиях.

Литература

1. Dai W., Toshimitsu H. Statistics in Medicine. 4rd ed. Oxford, UK: Academic Press, 2022. 795 p.

2. Lawrence R. K. The statistical approach to economics // Journal of Econometrics. 1988. Vol. 37. No 1. P. 7–26.

3. Zhan W., Rainer F., Fang A. Application Of Statistics In Engineering Technology Programs // American Journal of Engineering Education. 2010. Vol. 1. No 1. P. 65–78.

4. PySATL-Experiment [Электронный ресурс]: `URL:https://github.com/PySATL/pysatl-experiment` (дата обращения: 15.04.2025).

5. Epps T. W., Pulley L. B. A Test for normality based on the empirical characteristic function // Biometrika. 1983. Vol. 70. No 3. P. 723–726.

6. Kolmogorov A. N. Sulla determinazione empirica di una legge di distribuzione // Giornale dell'Istituto Italiano degli Attuari. 1933. Vol. 4. P. 83–91.

7. Mimoto N., Zitikis R. The Atkinson index, the Moran statistic, and testing exponentiality // Japan Statist. Soc. 2008. Vol. 38. P. 187–205.

8. Cox D. R., Oakes D. Analysis of survivial data. New York: Chapman and Hall, 1984. 212 p.

9. Bush M. G. A Modified Cramer-von Mises and Anderson-Darling test for the Weibull distribution with unknown location and scale parameters // Communications in Statistics-theory and Methods. 1983. Vol. 12. P. 2465–2476.

10. Deshpande J. V. A class of tests for exponentiality against increasing failure rate average alternatives // Biometrika. 1983. Vol. 70. P. 514–518.

11. Epstein B. Tests for the validity of the assumption that the underlying distribution of life is exponential (part 1) // Technometrics. 1960. Vol. 2. P. 83–107.

12. Frozini B. V. On the distribution and power of a goodness-of-fit statistic with parametric and nonparametric applications. Amsterdam-Oxford-New York: North-Holland, 1987. P. 133–154.

13. Gail M. H., Gastwirth J. L. A scale-free goodness-of-fit test for the exponential distribution based on the Gini statistic // J. R. Stat. Soc. Ser. B. 1978. Vol. 40. No 3. P. 350–357.

14. Gnedenko B. V., Belyayev Yu. K., Solovyev A. D. Mathematical Models of Reliability Theory. New York, London: Academic Press, 1969. 506 p.

15. Lin C. C., Mudhoikar G. S. A test of exponentiality based on the bivariate F distribution // Technometrics. 1980. Vol. 22. P. 79–82.

16. Hegazy Y. A. S., Green J. R. Some new goodness-of-fit tests using order statistics // Journal of the Royal Statistical Society. Series C (Applied Statistics). 1975. Vol. 24. P. 299–308.

17. Kimber A. C. Tests for the exponential, Weibull and Gumbel distributions based on the stabilized probability plot // Biometrika. 1985. Vol. 72. P. 661–663.

18. Kochar S. C. Testing exponenttality against monotone failure rate average // Communications in Statistics – Theory and Methods. 1985. Vol. 14. P. 381–392.

19. Gail M. H., Gastwirth J. L. A scale-free goodness-of-fit test for the exponential distribution based on the Lorenz curve // Journal of the American Statistical Association. 1978. Vol. 73. P. 787–793.

20. Tchirina A. V. Bahadur efficiency and local optimality of a test for exponentiality based on the Moran statistics // Journal of Mathematical Sciences. 2005. Vol. 127. No 1. P. 1812–1819.

21. Shapiro S. S., Wilk M. B. An analysis of variance test for the exponential distribution (complete samples) // Technometrics. 1972. Vol. 14. P. 355–370.

22. Hahn G. J., Shapiro S. S. Statistical Models in Engineering. New York: John Wiley and Sons, 1967. 355 p.

23. Wong P. G., Wong S. P. An extrernal quotient test for exponential distributions // Metrika. 1979. Vol. 26. P. 1–4.

24. PySATL-Criterion [Электронный ресурс]: `URL:https://github.com/PySATL/pysatl-criterion` (дата обращения: 15.04.2025).

УДК 519.683.8

Голощапова А. М.

Применение методов оптимизации гиперпараметров для повышения точности моделей прогнозирования кликов рекламы

Рекомендовано к публикации старшим преподавателем Кижаевой Н. А.

1. Введение. С развитием интернета все больше людей выбирают онлайн-услуги благодаря удобству и экономии времени. В стремлении привлечь потребителей к совершению покупок компании активно размещают рекламу на различных платформах.

В онлайн-рекламе показатель кликов рекламного объявления (CTR) является важным показателем для оценки эффективности кампании. Точность предсказания кликов влияет также на оптимальное распределение бюджета и стратегию размещения рекламы. Однако задача прогнозирования CTR сталкивается с рядом вызовов, таких как большой объем данных, высокая размерность признаков и дисбаланс классов, что требует применения современных методов машинного обучения и оптимизации [1, 2]. Одним из наиболее эффективных подходов к повышению точности моделей является оптимизация гиперпараметров [2]. Они контролируют процесс обучения модели и определяют ее способность к обобщению, избегая переобучения или, наоборот, недостаточного обучения. Применение методов оптимизации для подбора гиперпараметров, например, поиск по сетке (grid search), случайный поиск (random search), байесовская оптимизация (Bayesian optimization) и других, позволяет улучшить качество предсказаний. Цель работы – продемонстрировать, как выбор метода оптимизации гиперпараметров влияет на качество предсказания CTR. Работа имеет практическую значимость для специалистов в области анализа данных и маркетинга, так как предлагает подход к улучшению качества моделей прогнозирования CTR с использованием современных методов оптимизации.

Голощапова Алла Михайловна – студент, Санкт-Петербургский государственный университет; e-mail: st113272@student.spbu.ru, тел.: +7(993)446-47-36

2. Обзор данных. Для задачи используется датасет Avazu CTR Prediction, содержащий данные о показах мобильной рекламы. Датасет включает в себя как числовые, так и категориальные признаки, влияющие на вероятность клика пользователя.

Числовые признаки: hour – время суток, когда был совершен клик, banner pos (позиция баннера на странице), C1, C14 – C21 (признаки, скрытые для сохранения анонимности).

Категориальные признаки: site id (идентификатор сайта), site domain (домен сайта), site category (категория сайта), app id (идентификатор приложения), app domain (домен приложения), app category (категория приложения), device id (идентификатор устройства), device ip (IP-адрес устройства), device model (модель устройства). Целевая переменная: бинарная переменная click, равная 1, если был клик, и 0 – иначе.

3. Подготовка данных. Перед обучением модели данные были предварительно обработаны.

Числовые признаки приводятся к единому масштабу с помощью StandardScaler. Этот метод преобразует данные так, чтобы среднее значение каждого признака стало равно 0, а разброс данных (стандартное отклонение) – 1. Для этого из каждого значения вычитается среднее значение признака, а затем результат делится на стандартное отклонение.

Категориальные данные кодируются с помощью OneHotEncoder. Этот метод преобразует категориальные данные в бинарные столбцы для каждого возможного значения категориального признака. Также для кросс-валидации был использован StratifiedKFold, что обеспечило примерно одинаковое соотношение классов (клик/не клик) в каждом фолде. Для обучения модели данные были разделены на обучающую и валидационную выборки с использованием функции train_test_split из библиотеки scikit-learn.

Признак часа (hour) был преобразован в формат даты и времени. Из него были извлечены два признака: час дня (значение времени в пределах 24 часов) и день недели (число от 0 до 6, где 0 – понедельник).

Основная задача – предсказать вероятность клика $P(y = 1|\mathbf{x})$, где $\mathbf{x}$ – вектор признаков.

4. Модель и методы оптимизации гиперпараметров. В исследовании были использованы три модели машинного обучения.

4.1. LightGBM. Для задачи прогнозирования кликов на рекламу была выбрана модель LightGBM – алгоритм градиентного бустинга, который активно используется для обработки больших данных благодаря своей эффективности и высокой скорости обучения [1–3]. В модели LightGBM каждый новый элемент (дерево) строится на основе ошибок, сделанных предыдущими деревьями, что позволяет последовательно улучшать точность предсказаний [4, 5].

4.2. Методы оптимизации гиперпараметров. Для оптимизации гиперпараметров модели LightGBM использовались:

- метод случайного поиска [6];
- байесовская оптимизация, реализованная через библиотеки Python:
  - bayesian-optimization (на основе гауссовских процессов – Gaussian processes) [7];
  - Optuna (с алгоритмом древовидного парзеновского оценщика – Tree-structured Parzen Estimator (TPE)) [8, 9].

Метод случайного поиска выбирает случайные комбинации гиперпараметров из заданных диапазонов и оценивает их с помощью кросс-валидации [6]. Это позволяет значительно сократить количество проверяемых комбинаций по сравнению с полным перебором, когда проверяются все возможные значения.

Для байесовской оптимизации гиперпараметров использовалась библиотека bayesian-optimization, реализующая метод на основе гауссовских процессов. Подход последовательно анализирует результаты предыдущих испытаний, чтобы определять наиболее перспективные комбинации параметров для проверки [7]. Этот метод эффективен в случае, если возможных комбинаций гиперпараметров много, а вычислительные ресурсы позволяют потратить больше времени на поиск оптимальных значений.

Optuna использует альтернативный метод байесовской оптимизации для подбора оптимальных гиперпараметров. В ней реализован алгоритм древовидного парзеновского оценщика, который работает на основе моделирования вероятностей гиперпараметров, используя информацию из предыдущих испытаний [8, 9]. Алгоритм учитывает историю испытаний и сосредотачивается на наиболее перспективных областях пространства гиперпараметров, снижая количество итераций, в отличие от случайного поиска или перебора по сетке.

5. Результаты. Для оценки качества моделей машинного обучения при прогнозировании кликов использовались следующие метрики: Log loss – функция ошибки, применяемая в классификации, когда предсказывается вероятность принадлежности к классу. Чем ниже значение log loss, тем точнее модель.

Время работы модели – время, затраченное на обучение модели и настройку гиперпараметров. Результаты сравнения моделей на тестовых данных приведены в таблице.

Таблица. Сравнение методов оптимизации гиперпараметров

Метод	**Log loss**	**Время работы, с**
Без подбора параметров	0,3357	120,5
Случайный поиск	0,2983	188,5
Bayesian-optimization	0,2956	213,3
Optuna	0,2878	215,7

6. Выводы. Модель без оптимизации гиперпараметров показала наибольший log loss (0,3357), что подтверждает важность настройки параметров для улучшения точности модели.

Для оптимизации гиперпараметров модель случайного поиска показала уменьшение log loss до 0,2983, что свидетельствует о положительном влиянии случайного выбора гиперпараметров на производительность модели. Однако это улучшение не столь значительное, как в случае с более сложными методами оптимизации.

Модель байесовской оптимизации показала еще лучшее значение log loss: 0,2956, что подтверждает эффективность байесовского подхода для подбора гиперпараметров. Метод работает, предсказывая значения гиперпараметров на основе предыдущих испытаний.

Модель с использованием Optuna показала наилучший результат log loss: 0,2878, что подтверждает ее высокую эффективность в задаче оптимизации гиперпараметров. Optuna позволяет находить гиперпараметры, которые дают наиболее точные результаты при разумных вычислительных затратах, хотя время работы модели с этим методом было наибольшим среди протестированных подходов.

Таким образом, на основе полученных результатов можно заключить, что оптимизация гиперпараметров значительно повышает точность моделей машинного обучения. Optuna продемонстрировала наилучшие результаты, однако методы байесовской оптимизации (на основе гауссовских процессов) и случайного поиска также оказались эффективными, что делает их применимыми для задач с ограничениями по времени и вычислительным ресурсам.

Литература

1. Xia Z., Mao S., Bai J. et al. A novel integrated network with LightGBM for click-through rate prediction // Research Square 2021, preprint.

2. Сорокина С. О., Михайлова Е. Г. Оптимизация гиперпараметров алгоритмов градиентного бустинга // Процессы управления и устойчивость. 2021. Т. 8. № 1. С. 326–331.

3. Ци Д., Буре В. М. Исследование методов прогнозирования временных рядов для предсказания качества воздуха: объяснительный сравнительный анализ // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2024. Т. 20. № 2. С. 206–219.

4. Белоусов Ю. В., Полежаев С. А. Предсказание спроса с помощью методов машинного обучения // Процессы управления и устойчивость. 2019. Т. 6. № 1. С. 218–222.

5. Ke G., Meng Q., Finley T. et al. Lightgbm: a highly efficient gradient boosting decision tree // Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems, Long Beach, 4–9 December 2017. ВР. 3146–3154.

6. Bayesian Optimization [Электронный ресурс]: URL: https://github.com/bayesian-optimization/BayesianOptimization (дата обращения: 12.03.2025).

7. Bergstra J., Bengio Y. Random search for hyper-parameter optimization // The journal of machine learning research. 2012. No. 13. P. 281–305.

8. Optuna – A hyperparameter optimization framework [Электронный ресурс]: URL: https://optuna.org/ (дата обращения: 12.03.2025).

9. Akiba T., Sano S., Yanase T., Ohta T., Koyama M. Optuna: a next-generation hyperparameter optimization framework // KDD '19: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York: ACM. 2019. P. 2623–2631.

УДК 004.7:004.623:616-71

Жариков Д. С., Блеканов И. С.

Методы распределенной загрузки медицинских данных в формате КТ-снимков

1. Введение. Современные методы компьютерной томографии (КТ) в кардиологии позволяют получать детальные изображения сердца, дающие ценные диагностические данные для врачей. Внедрение нейросетевых моделей искусственного интеллекта (ИИ) в анализ КТ-снимков способствует повышению точности постановки предварительных диагнозов и сокращению времени, затрачиваемого на рутинные операции [1]. Однако при работе с медицинскими данными, особенно в условиях дистанционного взаимодействия специалистов, необходимо обеспечить:

1. Строгое соответствие нормативным требованиям РФ по защите персональных данных (ФЗ №152 и ФЗ №323) и криптографическим стандартам ГОСТ.

2. Возможность возобновляемой передачи тяжелых DICOM-файлов при сбоях, чтобы не повторять загрузку уже успешно переданного объема.

3. Исключение дублирования данных (особенно при многократной отправке одних и тех же КТ-снимков). Удобство и прозрачность процесса для врача и сервисов машинного обучения.

В данной работе рассматриваются методы распределенной загрузки медицинских данных в контексте медицинской информационной системы [2], решающей указанные задачи за счет организации безопасного канала передачи, правильного управления крупными файлами (DICOM) и интеграции с хранилищем на базе S3-протокола.

Жариков Даниил Сергеевич – магистрант, Санкт-Петербургский государственный университет; e-mail: st076580@student.spbu.ru, тел.: +7(812)363-60-00 (9553)

Блеканов Иван Станиславович – доцент, Санкт-Петербургский государственный университет; e-mail: i.blekanov@spbu.ru, тел.: +7(812)363-60-00 (9553)

2. Описание проблемы. В рамкам применения методов и технологий искусственного интеллекта для анализа медицинских данных одной из актуальных задач является удаленная загрузка КТ-снимков пациента в систему помощи принятия решения врачам [3]. Данный процесс должен обеспечить на стороне системы:

- доступ для просмотра, обсуждения и разметки данных врачом или другим специалистом;
- адаптируемость данных к использованию методами и технологиями искусственного интеллекта для первичной (автоматической) диагностики.

Для оперативной обработки данных (например, медицинских изображений) ИИ-моделями и технологиями требуется безопасная и эффективная среда передачи. Однако КТ-снимки, упакованные по стандарту DICOM [4] (со структурой, описываемой DICOMDIR-файлом), зачастую имеют значительный размер, при передаче таких данных через Интернет или локальную сеть в контуре клиники возникают следующие проблемы:

1. Неполная передача в результате прерываний соединения.
2. Дублирование данных.
3. Нарушение требований к защите протокола к передачи медицинских данных.

3. Анализ готовых решений. Важно изучить протоколы и методы защиты данных в масштабируемой архитектуре.

3.1. Протоколы передачи и безопасность. Внутри локального контура распространены протоколы SMB, NFS или SFTP, отличающиеся высокой скоростью обмена файлами. При работе через Интернет актуальнее HTTPS (TLS), WebDAV + TLS или gRPC поверх TLS. В контексте российского законодательства необходимо интегрировать отечественные криптографические алгоритмы (ГОСТ 34.12-2018, ГОСТ 34.10-2018).

3.2. Структура хранения DICOM. Классические PACS-системы (Picture Archiving and Communication System) предлагают

комплексные решения для хранения и анализа DICOM-изображений [5]. Однако они бывают избыточны, требуют сложной интеграции и не всегда полностью подстраиваются под локальные криптостандарты. Поэтому часто выбирают более гибкий подход:

- парсер DICOMDIR для извлечения иерархических связей (пациент > исследование > серия > снимок);
- объектное S3-хранилище (например, MinIO), где файлы хранятся в псевдо-древовидной структуре с разделением по идентификаторам пациентов и исследований.

Такой способ позволяет легко проверять наличие дубликатов и динамически масштабировать объем хранилища (при росте числа пациентов и снимков).

4. Описание решения. Была разработана концепция, учитывающая надежность, пропускную способность и удобство работы с большими медицинскими снимками.

4.1. Архитектура клиентской части. Ключевой элемент – веб-приложение врача, где пользователь выбирает локальную папку с DICOMDIR-файлом и снимками. Система автоматически:

1. Отправляет DICOMDIR-файл на бэкенд (через WebDAV + HTTPS), чтобы выяснить, какие снимки уже есть в хранилище (MinIO) и какие необходимо дозагрузить.
2. Делит файлы на небольшие «порции» (chunk), при сбое соединения повторно запрашивает «недостающие части».
3. По окончании загрузки уведомляет ИИ-сервис для запуска анализа.

Используемые технологии:

- Nginx/WebDAV+TLS (ГОСТ) дает защищенный канал, поддерживающий PUT/DELETE/PROPFIND и алгоритмы ГОСТ.
- MinIO (S3) предоставляет структуру для хранения DICOM-файлов по иерархии (пациент/исследование/серия).

4.2. Алгоритм распределенной загрузки медицинских данных. На рис. 1 представлена диаграмма последовательностей,

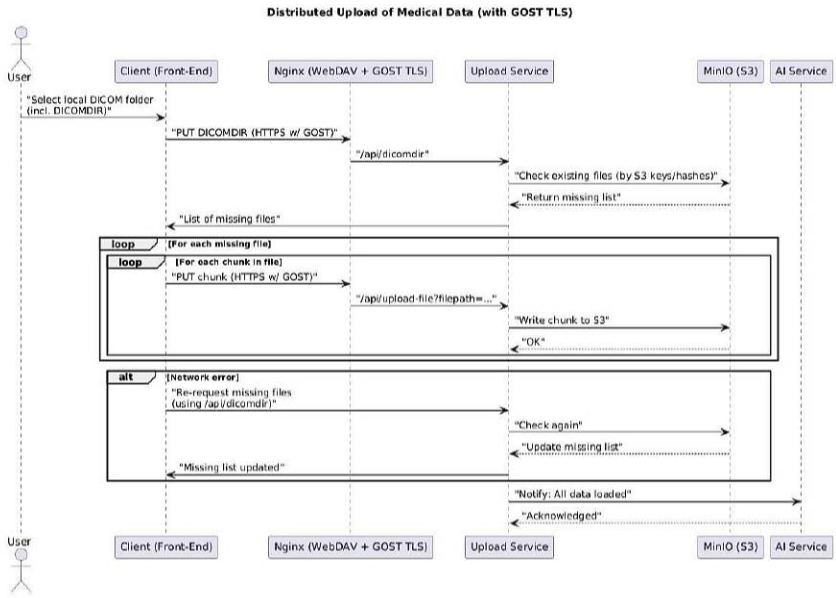


Рис. 1. Диаграмма последовательностей

отражающая взаимодействие пользователя, клиентского приложения, сервиса загрузки (бэкенд) и S3-хранилища MinIO.

4.2.1. Описание диаграммы. Пользователь в веб-интерфейсе выбирает локальный каталог или ZIP-архив с DICOM-файлами и DICOMDIR. Затем клиентское приложение отправляет DICOMDIR на сервис, где происходит сопоставление со структурой, уже хранящейся в MinIO. Сформированный список недостающих файлов передается клиенту, который загружает их частями. При сетевом сбое клиент снова запрашивает список, исключая уже сохраненные части, и продолжает передачу. По окончании загрузки сервис уведомляет модуль ИИ о готовых данных.

4.2.2. Псевдокод алгоритма распределенной загрузки

```
1. user_input := select_local_directory()
2. SEND DICOMDIR to backend over TLS:
   missing_files := backend.analyze_dicomdir()
3. FOR each file IN missing_files:
```

```
    chunk_list := chunkify(file, chunk_size=5MB)
    FOR each chunk IN chunk_list:
        TRY:
           SEND chunk to backend
        CATCH network_exception:
           missing_files := backend.analyze_dicomdir()
           REPEAT from last untransferred chunk
4. ON completion of all files:
   backend.notify_AI_service()
   show_message("Upload completed, AI analysis started")
```

Реализуется *распределенная* модель загрузки, так как части DICOM-файла могут передаваться из разных клиентских узлов, а сервер параллельно обрабатывает их, сохраняя в S3.

5. Тестирование системы. Для проверки решения нужен эксперимент, отражающий реальные условия передачи и хранения данных.

5.1. Постановка эксперимента. Была развернута экспериментальная среда (Docker-контейнеры с Nginx+WebDAV+TLS + MinIO) на сервере (4 CPU, 8 ГБ ОЗУ). Пропускную способность, задержки и обрывы (2–5 %) моделировали с помощью `tc netem` и `tc qdisc loss random`. Часть файлов заранее находилась в MinIO для проверки дубликатов. Для теста загрузили 50 DICOM-файлов (1 ГБ каждый) в следующих сценариях: локальная сеть (без шифрования и с TLS+ГОСТ), Интернет (∼100 Мбит/с) и CDN (см. таблицу).

5.2. Результаты эксперимента. В ходе эксперимента фиксировалась скорость передачи, количество сетевых сбоев и итоговое время загрузки. Обобщенные данные приведены в таблице.

Таблица. Сценарии тестирования и результаты

Сценарий	*Сеть*	*Шифрование*	*Скорость (МБ/с)*	*Сбои*	*Время (с)*
Локальная сеть (SMB)	1 Гбит/с	Нет	80–90	0	∼12
Локальная сеть (WebDAV+TLS)	1 Гбит/с	ГОСТ	70–75	0	∼14
Интернет (WebDAV+TLS)	∼100 Мбит/с	ГОСТ	8–10	2	∼120
Интернет (WebDAV+TLS +CDN)	∼100 Мбит/с	ГОСТ	11–12	2	∼90

Эксперимент показал, что даже при 2–5 % случайных обрывах сеть эффективно восстанавливала передачу. Часть файлов уже находилась в хранилище, поэтому система не загружала дубликаты, экономя до 25–30 % трафика и времени.

6. Заключение. В работе предложена архитектура распределенной системы загрузки и обработки медицинских данных, включающая безопасный канал (WebDAV+TLS с ГОСТ-алгоритмами); алгоритм дозагрузки «порций» с проверкой дубликатов; автоматическое уведомление ИИ-модуля о готовности снимков.

Проведенный эксперимент показал, что предлагаемая схема исключает дубли, устойчива к обрывам и обеспечивает достаточно высокую скорость передачи (8–90 МБ/с). В дальнейшем планируется оптимизировать разбиение на «порции», расширить логирование данных и интеграцию с PACS- и ML-сервисами.

Литература

1. Коваленко Л. А., Ларин Е. С., Блеканов И. С., Ким Г. И. Методы построения 3D модели аорты сердца и ее сечения для автоматизации процесса КТ-диагностики // Процессы управления и устойчивость. 2023. Т. 10. № 1. С. 298–303.

2. Фролов С. В., Маковеев С. Н., Семенова С. В. Современные особенности развития медицинских информационных систем // Врач и информационные технологии. 2010. № 2. С. 4–9.

3. Ежов Ф. В., Коваленко Л. А., Разумилов Е. С., Блеканов И. С. Инструменты краудсорсинга для анализа и обработки медицинских изображений в виде снимков КТ // Процессы управления и устойчивость. 2023. Т. 10. № 1. С. 291–297.

4. Larobina M., Murino L. Medical image file formats // Journal of digital imaging. 2014. Vol. 27. P. 200–206.

5. Mustra M., Delac K., Grgic M. Overview of the DICOM standard // 50th international symposium ELMAR. IEEE 2008. Vol. 1. P. 39–44.

УДК 004.318

Зубрицкий И. В.

Эффективный по занимаемой площади метод реализации аппаратной поддержки криптографической защиты информации

Рекомендовано к публикации профессором Богдановым А. В.

1. Введение. Что дает аппаратная поддержка шифрования? Во-первых – скорость: сравнивая более производительный intel core i7-870 с менее производительным intel core i5-661, в котором появилась аппаратная поддержка шифрования AES, видно, что аппратное шифрование заметно быстрее (см. рис. 1). Во-вторых – экономия энергии: используя меньше операций для выполнения шифрования, снижается потребление энергии. В-третьих – защита от атак: программное выполнение шифрования использует много вызовов, которые могут быть использованы для атаки по побочным каналам и определению ключа шифрования, но аппаратные блоки шифрования делают такие атаки менее эффективными.

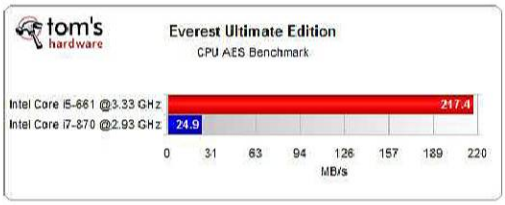


Рис. 1. Тесты шифрования AES в пакете Everest Ultimate Edition

Есть два вида шифрования: симметричное и асимметричное. Главное различие между ними в том, что в симметричном шифровании используется один и тот же ключ для шифрования и дешифрования, а в асимметричном – разные. Асимметричные шифры, такие как RSA или DSA, обычно используются для установления соединения и создания ключа для симметричного шифрования. Они не

Зубрицкий Иван Владимирович – магистрант, Московский физико-технический институт; e-mail: zubritskii.iv@phystech.edu, тел.: +7(977)638-97-18

используются для обработки больших объемов данных. Симметричные шифры, такие как ГОСТ 34.12 или AES, наоборот, используются для передачи больших объемов информации, например текста, аудио или видео. Поэтому важно, чтобы алгоритмы симметричного шифрования работали быстро.

Исходя из предназначения алгоритмов была поставлена задача реализовать оптимизированные по площади модули аппаратного ускорения криптографических алгоритмов зашифрования и расшифрования AES и зашифрования ГОСТ 34.12-2015 «Кузнечик».

2. AES (NIST FIPS 197). Шифр AES является блочным, симметричным с блоком размером 128 бит. Типовой раунд шифрования и расшифрования состоит из четырех операций: SubBytes – нелинейная замена байт, ShiftRows – сдвиг рядов в матрице значений State, MixColumns – линейное преобразование столбца в State, AddRoundKey – XOR с раундовым ключом. Раунд расшифрования представляет собой обратный порядок обратных операций. Алгоритм раунда расшифрования можно привести к виду, похожему на раунд зашифрования. Так как функции SubWord и ShiftRows работают с отдельными байтами, а значит они коммутативны, вследствие чего можно поменять их местами. Функции AddRoundKey и MixColumns являются линейными, следовательно по свойствам линейности можно поменять их местами, если заранее провести операцию MixCloumns с ключом (см. рис. 2).

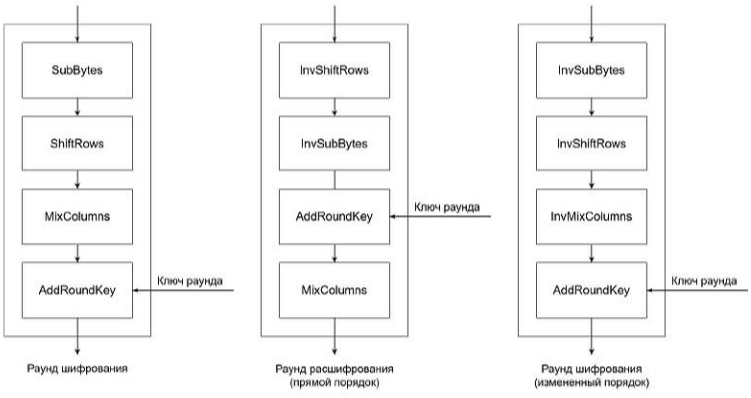


Рис. 2. Раунд шифрования и расшифрования AES

2.1. Линейная операция MixColumns. Данная операция является умножением матрицы 4×4 байта на столбец из четырех байт в поле Галуа $\mathrm{GF}(2^8)$ (см. рис. 3) [1]. Реализация операции умножения в поле Галуа в аппаратуре является нетривиальной задачей, так как нахождение остатка от деления является многоэтапной функцией. Вследствие того, что в операции MixColumns проводится произведение на константы, возможно найти оптимизации данной функции. В данной работе для всех линейных преобразований, задействующих умножение на константы в поле Галуа, была использована оптимизация, описанная в статье [2]. При произведении в поле Галуа, элементам поля $\mathrm{GF}(2^m)$ можно поставить в соответствие матрицы элементов поля $\mathrm{GF}(2)$ размера $m \times m$, m-битным переменным в соответствие ставятся столбцы размером $m \times 1$. Произведение на такие матрицы использует битовые логические операции AND и XOR и не требует нахождения остатка от деления на порождающий многочлен и из-за этого является предпочтительным как для понимания разработчика, так и для уменьшения площади, занимаемой модулем аппаратной поддержки. Используя данную оптимизацию, матрица преобразования InvMixColumns 4×4 байт преобразуется в матрицу 32×32 бит. Преобразование MixColumns было реализовано стандартным алгоритмом произведения в поле Галуа, так как матрица этого преобразования состоит из малых констант, умножение на которые является достаточно простой операцией.

MixColumns

$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix}$$

InvMixColumns

$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{bmatrix} 0e & 0b & 0d & 09 \\ 09 & 0e & 0b & 0d \\ 0d & 09 & 0e & 0b \\ 0b & 0d & 09 & 0e \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix}$$

Рис. 3. Операции MixColumns и InvMixColumns

2.2. Нелинейная операция SubBytes. Эта операция задает нелинейную замену байтовых подблоков по таблице замен. В документации NIST FIPS 197 [1] описан алгоритм формирования таблицы замен:

1. Нахождение обратного элемента в поле $\mathrm{GF}(2^8)$ (нулевому элементу в соответствие ставится нулевой элемент).

2. Линейное преобразование умножением на матрицу и XOR с константой 0×63.

Нелинейная операция расшифрования InvSubBytes также задает нелинейную замену байтовых подблоков по таблице замен. Эта операция будет обратной к SubBytes, и ее таблица будет формироваться по аналогичному алгоритму:

1. Линейное преобразование умножением на обратную матрицу и XOR с константой 0×63.
2. Нахождение обратного элемента в поле $\mathrm{GF}(2^8)$ (нулевому элементу в соответствие ставится нулевой элемент).

Реализация этих операций в виде блоков табличной замены крайне затратна по площади, поэтому необходимо оптимизировать его, сделав декомпозицию самой сложной части – нахождения обратного в поле $\mathrm{GF}(2^8)$. На данный момент самой эффективной декомпозицией нахождения обратного в поле $\mathrm{GF}(2^8)$ является переход к составному полю GF (2^8)/GF(2^4)/GF(2^2)/GF(2) [3,4]. В таком поле 8-битное число представляется как два 4-битных, 4-битное как два 2-битных, а 2-битное представляется как два 1-битных. Нахождение же обратного в поле GF(2) задача тривиальная и не требующая затрат по площади, так как обратным является само число. Обратные в полях степени 2, 4 и 8 будут вычисляться как линейные комбинации их подблоков, по следующим формулам:

$$q, q' \in GF(2^8), \quad \gamma_1, \gamma_0, \gamma_1', \gamma_0' \in GF(2^4),$$

$$\begin{cases} q = \gamma_1 y + \gamma_0, \\ q' = \gamma_1' y + \gamma_0'. \end{cases} \quad r(y) = y^2 + \tau y + \gamma,$$

$$\begin{cases} \gamma_1' = (\gamma_1^2 \nu + \gamma_1 \gamma_0 \tau + \gamma_0^2)^{-1} \gamma_1, \\ \gamma_0' = (\gamma_1^2 \nu + \gamma_1 \gamma_0 \tau + \gamma_0^2)^{-1} (\gamma_0 + \gamma_1 \tau). \end{cases}$$

$$\gamma, \gamma' \in GF(2^4), \quad \Gamma_1, \Gamma_0, \Gamma_1', \Gamma_0' \in GF(2^2),$$

$$\begin{cases} \gamma = \Gamma_1 z + \Gamma_0, \\ \gamma' = \Gamma_1' z + \Gamma_0'. \end{cases} \quad s(z) = z^2 + Tz + N,$$

$$\begin{cases} \Gamma_1' = (\Gamma_1^2 N + \Gamma_1 \Gamma_0 T + \Gamma_0^2)^{-1} \Gamma_1, \\ \Gamma_0' = (\Gamma_1^2 N + \Gamma_1 \Gamma_0 T + \Gamma_0^2)^{-1} (\Gamma_0 + \Gamma_1 \tau). \end{cases}$$

$$\Gamma, \Gamma' \in GF(2^2), \quad g_1, g_0, g_1', g_0' \in GF(2^2),$$
$$\begin{cases} \Gamma = g_1 w + g_0, \\ \Gamma' = g_1' w + g_0'. \end{cases} \quad t(w) = w^2 + w + 1,$$

$$\begin{cases} g_1' = (g_1^2 + g_1 g_0 + g_0^2)^{-1} g_1, \\ g_0' = (g_1^2 + g_1 g_0 + g_0^2)^{-1} (g_0 + g_1). \end{cases}$$

Учитывая, что в $GF(2) : g^2 = g^{-1} = g$, получим

$$\begin{cases} g_1' = g_1, \\ g_0' = g_1 + g_0. \end{cases}$$

Таким образом, разбивая сложную задачу по поиску обратного в GF(2^8) на более простые задачи, исходя из [3] и сравнения синтеза написанных RTL версий, возможно сократить площадь модуля, реализующего операцию SubBytes и InvSubBytes на 60%.

3. ГОСТ 34.12-2018 "Кузнечик". «Кузнечик» – криптографический алгоритм симметричного шифрования с блоком данных 128 бит. Шифр состоит из 10 раундов, которые представляют собой связку из трех функций: сложения с ключом в поле GF(2), линейное L преобразование в поле GF(2^8) и табличную байтовую замену S.

3.1. Линейное L преобразование. Линейное преобразование L состоит из 16 преобразований l, которое является байтовым сдвигом вправо с заменой старшего байта на линейную комбинацию байтов, вычисленную в поле Галуа GF(2^8).

$$l(a_{15}, \ldots, a_0) = \bigtriangledown(148 \cdot \Delta(a_{15}) + 32 \cdot \Delta(a_{14}) + 133 \cdot \Delta(a_{13}) + 16 \cdot \Delta(a_{12}) + \\ + 194 \cdot \Delta(a_{11}) + 192 \cdot \Delta(a_{10}) + 1 \cdot \Delta(a_9) + 251 \cdot \Delta(a_8) + 1 \cdot \Delta(a_7) + 192 \cdot \Delta(a_6) \\ + 194 \cdot \Delta(a_5) + 16 \cdot \Delta(a_4) + 133 \cdot \Delta(a_3) + 32 \cdot \Delta(a_2) + 148 \cdot \Delta(a_1) + 1 \cdot \Delta(a_0)).$$

Так же, как и в криптографическом алгоритме AES, операции проводятся в поле Галуа $GF(2^8)$, но с отличным порождающим полиномом. Несмотря на это, способ оптимизации остается схожим. Каждое преобразование l можно представить в виде умножения в поле GF(28) на матрицу размером 16×16 байт. Таким образом, возведя эту матрицу в 16 степень, можно получить матрицу всего преобразования L (см. рис. 4). Далее аналогично оптимизации линейных функций алгоритма AES, можно заменить элементы поля $GF(2^8)$ на битовые матрицы 8×8, получив таким образом матрицу 128×128 бит, умножение на которую осуществлять просто и с алгоритмической, и с аппаратной точки зрения [2].

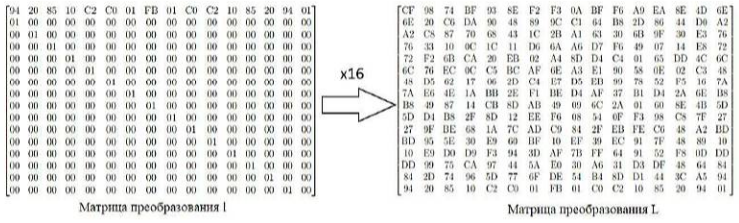


Рис. 4. Матрицы преобразований l и L шифрования «Кузнечик»

3.2. Нелинейное преобразование S. Нелинейная операция S задает нелинейную замену байтовых подблоков по таблице замен S-box П. Разработчики отечественного алгоритма «Кузнечик» объявили, что таблица замены была сформирована случайным образом, поэтому официального математического алгоритма, который можно было бы оптимизировать, не существует [5]. Однако в исследовании [6] был найден алгоритм преобразования байта, результат которого полностью соответствует таблице замен (см. рис. 5). В данной схеме присутствуют четыре основных типа функций. Первый – табличные замены полубайтовых блоков $\gamma_0, \gamma_1, \varphi, \sigma$; второй – произведение в поле Галуа $GF(2^4)$, обозначенное точкой; третий – поиск обратного в поле Галуа $GF(2^4)$, обозначенный буквой I; и последний – линейные преобразования в виде произведения на матрицу 8×8 бит в поле GF(2), обозначенные как α и ω. Основываясь на этом алгоритме [6], была проведена декомпозиция табличной замены криптографического алгоритма «Кузнечик». Проведя сравнение синтеза двух написанных RTL-описаний: версию с декомпозицией и версию

с табличным поиском, было вычислено, что сокращение аппаратных расходов составило 40%.

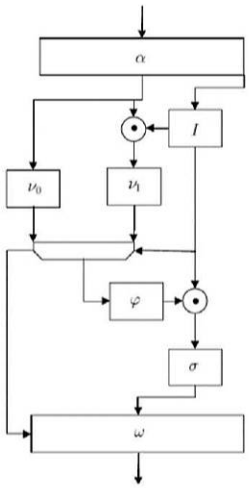


Рис. 5. Схема декомпозиции S-box П

4. Результаты. Главный результат модулей аппаратного шифрования: увеличение скорости шифрования. По расчетам, скорость обработки блока данных, измеренная в тактах должна увеличиться в 25 раза для алгоритма «Кузнечик» и в 16 раз для алгоритма AES. Аппаратное шифрование привнесло бы в систему команд следующие операции. Для алгоритма AES: команда начала раунда зашифрования, совершающая нелинейную замену байт и байтовый сдвиг; команда конца раунда зашифрования, совершающая линейное преобразование; команда начала раунда расшифрования, совершающая обратную нелинейную замену байт и обратный байтовый сдвиг; команда конца раунда зашифрования, совершающая обратное линейное преобразование. Для алгоритма «Кузнечик»: команда начала раунда шифрования, совершающая нелинейную замену байт S; команда конца раунда, совершающая линейное преобразование L.

Литература

1. FIPS 197. Advanced Encryption Standard (AES) [Электронный ресурс]: URL:https://www.jedi-sec.com/downloads/Agencies/NIST/NIST.FIPS.197-upd1.pdf (дата обращения 01.06.25).

2. Курганов Е. А. О глубине аппаратной реализации блочного шифра Кузнечик // Интеллектуальные системы. Теория и приложения. 2016. Т. 20. Вып. 1. С. 61–78.

3. Canright D. A very compact S-box for AES // Rao J. R., Sunar B., Cryptographic Hardware and Embedded Systems, CHES 2005. Heidelberg: Springer Berlin. 2005. P. 441–455.

4. Satoh A., Morioka S., Takano K., Munetoh S. A compact rijndael hardware architecture with S-box optimization // Boyd C., Advances in Cryptology, ASIACRYPT 2001. Heidelberg: Springer Berlin. 2001. P. 239–254.

5. Shishkin V., Marshalko G. A memo on Kuznyechik S-box // ISO/IEC JTC 1/SC 27/WG 2 Cryptography and security mechanisms Convenorship: JISC, Japan. 2018. Officer's Contribution № 1804.

6. Biryukov A., Perrin L., Udovenko A. The secret structure of the S-box of Streebog, Kuznechik and Stribob // Cryptology ePrint Archive. 2015.

УДК 004.75

Ибатуллин Е. В., Ламзин Н. А.

Архитектура Data Mesh для среды больших данных

Рекомендовано к публикации профессором Богдановым А. В.

1. Введение. Область хранения данных на сегодняшний день является критической для множества отраслей. Это обусловило колоссальный рост объемов данных, требующих продуманную архитектуру и предобработку, удобную для конечного потребителя. В данной работе рассматривается задача интеграции нескольких систем хранения в одну среду данных. Также предложен вариант архитектуры data-mesh, нацеленный на решение данной задачи с использованием актуальных технологий распределенного хранения данных.

2. Обзор существующих методов. Для решения проблемы в области хранения данных многие компании разрабатывают свои собственные системы хранения. Также активно развиваются open-source решения. Рассмотрим современные подходы к проектированию систем хранения данных.

2.1. Существующие подходы. В связи с расширением объема информации, которую требуется обработать, стали появляться модели распределенных вычислений. В работе [1] показано, что основным аспектом такой системы является ее масштабируемость не вертикально, а горизонтально. С появлением таких подходов возникло понятие data-warehouse в классическом представлении, описанное в работе [2], которое стало следующим этапом после реляционных баз данных, предлагающее хранение структурированных данных. Но дальнейший рост мощностей, а также появление новых областей, таких как искусственный интеллект и «интернет вещей» показало, что такая система все еще не способна масштабироваться до нужных размеров. Слабое место находилось в слое предобработки, необходимом для табличных хранилищ. Решением стали подходы, подобные

Ибатуллин Евгений Вадимович – студент, Санкт-Петербургский государственный университет; e-mail: st113398@student.spbu.ru, тел.: +7(919)633-41-62

Ламзин Никита Андреевич – студент, Санкт-Петербургский государственный университет; e-mail: st096510@student.spbu.ru, тел.: +7(913)885-06-10

описанным в [3], где рассматривалась архитектура data-lake и причины перехода бизнеса на системы хранения данного типа. Решение заключалось в отказе от предобработки, что значительно ускорило процессы доставки данных. Однако это потребовало более высокого качества исходных данных и обязательного ведения метаданных. На сегодняшний день это считается классическим решением для среды больших данных. Но при дальнейшем масштабировании стало видно, что возникают проблемы нового типа. Интегрировать хранилища разнородных данных из разных областей становится новой передовой задачей перед системами хранения. В работе [4] показано, что виртуализация является необходимой частью для решения поставленной задачи. Следствием этого становится переход к архитектуре data-mesh, предложенной в [5]. Исследование развивает концепцию data-mesh через предложение практического способа ее реализации.

2.2. Актуальные решения. В основе современных технологий больших данных лежит система, предложенная благотворительной организацией Apache: Hadoop Distributed File System, архитектура которой описана в [6]. Эта система лежит в основе многих технологий хранения больших данных, а также используется для построения хранилищ типа data-lake. Также важно рассмотреть Apache Spark – фреймворк с открытым исходным кодом для реализации распределенной обработки неструктурированных и слабоструктурированных данных. Из описания к системе Apache Spark [7], видно, что решение предполагает потоковую обработку данных на вычислительном кластере. Описанные выше системы объединяются для реализации современных процессов big data.

3. Архитектура data-mesh. Проблема, которую не могут удовлетворить современные решения заключается в интеграции нескольких систем data-lake. Она возникает в результате проектирования бизнесом некоторых областей данных, необходимых для определенной отрасли. Для задач нового типа это является преградой к построению полной картины данных.

Предлагаемое решение подразумевает промежуточный слой, объединяющий области данных. Виртуализация проходит по аналогии с системой, предложенной в работе [4]. Для начала рассмотрим локально одну из областей данных. Такая среда представляет собой хранилище, в нашем случае это data-lake. В данное окружение поступают данные из некоторых потоков, и полученная информация

используется отделом для работы. Для реализации локальной виртуализации необходимо представить распределенные хранилища как единое целое, используя метаданные. Также требуется организация высокоскоростной коммуникации узлов системы хранения. Эта система подробно описана в [8], где представлена реализация с применением виртуального суперкомпьютера, использование облачных технологий и показана ее эффективность.

После виртуального представления каждой локальной области данных требуется объединить их, чтобы получить полноценную систему data-mesh, абстрактное представление которой предложено в [5]. На практике предлагается использовать промежуточный слой. Виртуальное представление позволяет организовать области данных как узлы кластера. На слое, который лежит между потребителем и системой данных, организуются наборы метаданных для определения положения данных в соответствие с каждым виртуальным узлом. Каждый узел имеет представление для своего локального кластера и, следовательно, определение расположения данных можно организовать через обращение к этому кластеру. Когда расположение данных определено, требуется извлечь данные из узлов систем хранения. Для этого используется высокопараллельная система извлечения Apache Spark, описанная выше. В итоге данную систему, начиная от локальных data-lake и заканчивая вычислительным кластером для извлечения данных, можно виртуализировать, в результате чего конечный пользователь получает единую систему хранения данных. Это и будет абстрактный слой, который работает с метаданными и системой извлечения для предоставления пользователю единого виртуализированного представления всей сложной системы. На рис. 1 изображена архитектура предложенной системы.

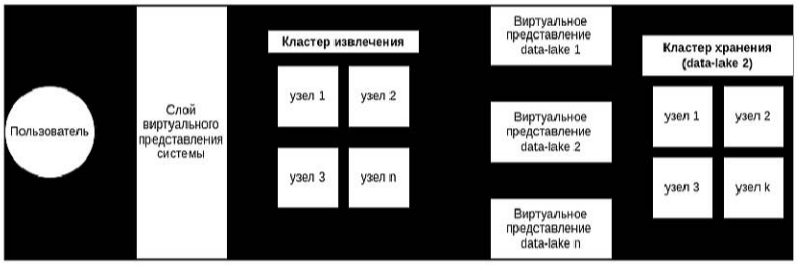


Рис. 1. Архитектура data-mesh

4. Заключение. Проходя основные этапы развития систем хранения информации, возникает потребность в интеграции нескольких архитектур данных. Архитектура data-mesh позволяет добиться этого, благодаря совместному использованию уже предложенных систем. В данной работе описаны основные технологии, необходимые для создания такой архитектуры. Предложен шаблон проектирования такой системы, основанный на методах, описанных для распределенных систем и виртуализации. Архитектура data-mesh является логическим продолжением развития систем хранения и направлена на интеграцию разнородных данных.

Литература

1. Кобышев С. С., Корхов В. В. Построение модели распределенной вычислительной системы // Процессы управления и устойчивость. 2016. Т. 3. № 1. С. 404–408.
2. Data-warehouse architecture [Электронный ресурс]: URL:https://cloud.google.com/learn/what-is-a-data-warehouse (дата обращения: 11.02.25).
3. Data-lake architecture [Электронный ресурс]: URL:https://yandex.cloud/ru/docs/glossary/datalake (дата обращения: 11.02.25).
4. Bogdanov A., Degtyarev A., Shchegoleva N., Khvatov V., Korkhov V. Principles of Big Data Virtualization // Computational Science and Its Applications (ICCSA). 2020. Vol. 12254. P. 67–81.
5. How to move beyond a monolithic data lake to a distributed data mesh [Электронный ресурс]: URL:https://martinfowler.com/articles/data-monolith-to-mesh.html (дата обращения: 11.02.25).
6. Hadoop distributed file system [Электронный ресурс]: URL:https://hadoop.apache.org/docs/r1.2.1/hdfs_design.html (дата обращения: 11.02.25).
7. Apache spark [Электронный ресурс]: URL:https://spark.apache.org/docs/latest/index.html (дата обращения: 11.02.25).
8. Gankevich I., Korkhov V., Bogdanov A., Degtyarev A. et al. Constructing virtual private supercomputer using virtualization and cloud technologies // Computational Science and Its Applications (ICCSA). 2014. Vol. 8584. P. 341–354.

УДК 004.032.26

Измайлов Н. О.

Обучение бинарной нейронной сети с использованием ее непрерывного аналога

Рекомендовано к публикации профессором Аббасовым М. Э.

1. Введение. Искусственный интеллект (ИИ) активно развивается, но традиционные нейронные сети требуют много вычислительных ресурсов, что затрудняет их использование на устройствах с ограниченными ресурсами и использование в целом. Бинарные нейронные сети (БНС), где веса и активации принимают только два значения, значительно снижают вычислительные затраты и потребление памяти, однако теряют точность [1]. Разработка методов обучения, обеспечивающих баланс между точностью и эффективностью, особенно с использованием непрерывных аналогов БНС, – ключ к созданию энергоэффективных ИИ-систем. Данная работа предлагает новый подход к интерпретации связи между БНС и их непрерывным аналогом.

2. Аналитический обзор. Под бинарной нейронной сетью понимается нейронная сеть, которая использует два значения (например -1 и 1) для представления данных, таких как веса и активации. Следуя [2], опишем процесс бинаризации весов и активаций.

Основная операция в глубоком обучении может быть выражена математически в виде

$$z = \mathbf{w}^{\mathrm{T}}\mathbf{a}, \tag{1}$$

где $\mathbf{w}, \mathbf{a} \in \mathbb{R}^n$ – вектор весов слоя нейронной сети и вектор активаций с предыдущего слоя соответственно. В БНС используется бинарное представление векторов

$$Q_x(\mathbf{x}) = \alpha \mathbf{B}_{\mathbf{x}},$$

где $\alpha \in \mathbb{R}^+, \mathbf{B}_{\mathbf{x}} \in \{-1, 1\}^n, n$ – размерность $\mathbf{x}$. Как правило, α и $\mathbf{B}_{\mathbf{x}}$ находятся из задачи минимизации ошибки квантизации:

Измайлов Никита Олегович – студент, Санкт-Петербургский государственный университет; e-mail: nikitaizmaylovv@yandex.ru, тел.: +7(927)334-35-47

$$\alpha^*, \mathbf{B}_{\mathbf{x}}^* \in \arg \min_{\alpha \in \mathbb{R}^+, \mathbf{B}_{\mathbf{x}} \in \{-1,1\}^n} \|\mathbf{x} - \alpha \mathbf{B}_{\mathbf{x}}\|_2^2 . \tag{2}$$

В [3] показано, что решением задачи (2) являются

$$\alpha^* = \frac{\|\mathbf{x}\|_1}{n}, \quad \mathbf{B}_{\mathbf{x}}^* = \text{sign}(\mathbf{x}), \tag{3}$$

где под $\text{sign}(\mathbf{x})$ подразумевается вектор, полученный из $\mathbf{x}$ поэлементным применением sign функции. Таким образом, основная операция в БНС может быть выражена в виде

$$z = Q_w(\mathbf{w})^{\mathrm{T}} Q_a(\mathbf{a}) = \alpha_{\mathbf{w}} \alpha_{\mathbf{a}} \left(\text{sign}(\mathbf{w})^{\mathrm{T}} \text{sign}(\mathbf{a})\right) . \tag{4}$$

Далее под бинарной сетью и ее непрерывным аналогом будем подразумевать сети с операциями (4) и (1), соответственно.

Благодаря выражению (4) выявляются преимущества использования двоичных значений – значительное уменьшение объема занимаемой памяти и использование бинарных операций XNOR и popcount в формуле (4) вместо стандартного умножения и сложения 32-битных чисел. Значит, БНС обеспечивает значительное ускорение производительности.

Классическое обучение нейронной сети включает прямой и обратный проходы: данные проходят через сеть от начала до выхода (прямой проход), затем рассчитывается ошибка и на ее основе, с помощью градиента, обновляются веса. Однако в БНС использование sign функции приводит к обнулению производных, из-за чего веса не могут обновляться. Поэтому научным сообществом разрабатываются различные подходы для обучения БНС [1], среди которых замена функции sign ее гладкой аппроксимацией [2], модификация функции потерь с целью снижения потерь точности [4], использование двухэтапных и иных схем обучения [5], адаптация архитектуры сети, учитывающая специфику бинарной задачи [5].

3. Связь между непрерывной и бинарной нейронными сетями. В данном исследовании предлагается метод получения оптимальных весов БНС, основанный на учете связи между непрерывной и бинарной сетями. Заметим, что функция потерь БНС является кусочно-постоянной функцией с разрывами второго рода по границам областей постоянства. Значения функции потерь БНС в областях постоянства соответствуют значениям функции потерь

непрерывной сети в бинарных точках. Таким образом, поверхность функции потерь БНС представляет собой множество параллельных разноуровневых «кусков» плоскостей («площадок»). Обучить БНС (определить оптимальные бинарные веса) – значит найти самую низкую площадку и соответствующий ей набор бинарных весов. Эта интерпретация объясняет те проблемы, с которыми сталкиваются исследователи при обучении БНС: затухание градиента, осциллирующие бинарные веса при обучении.

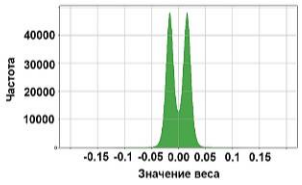


Рис. 1. График распределения весов БНС VGG-Small после обучения

Чтобы избежать всех указанных проблем при поиске оптимальных весов БНС, можно обратить внимание на то, что они являются решением следующей задачи условной оптимизации для функции потерь L:

$$\begin{cases} L(\mathbf{w}) \to \min, \\ \mathbf{w} \in \{-1, 1\}^n. \end{cases} \tag{5}$$

Задачу (5) можно решать методом штрафов, для чего введем функцию

$$\overline{\phi}(w_1, \ldots, w_n) = \sum_{i=1}^{n} (w_i - 1)^2 (w_i + 1)^2, \quad \mathbf{w} = (w_1, \ldots, w_n)^T.$$

Однако, следуя формуле (3), добавим масштабирующий множитель α в функцию ϕ, уменьшающий ошибку бинаризации:

$$\phi(\mathbf{w}, \alpha) = \sum_{i=1}^{n} (w_i - \alpha)^2 (w_i + \alpha)^2,$$

где масштабирующие множители будут считаться по формуле (3) для каждого канала отдельно.

Пусть $\lambda \geqslant 0$ – некоторое положительное число. Тогда задачу условной минимизации (5) можно заменить задачей безусловной минимизации

$$\overline{L}_\lambda(\mathbf{w}, \alpha) = L(\mathbf{w}) + \lambda \phi(\mathbf{w}, \alpha) \to \min_{\mathbf{w} \in \mathbb{R}^n, \alpha \in \mathbb{R}^+}.$$

В связи с вышеизложенными идеями, предложим алгоритм нахождения оптимальных весов БНС. Для удобства опишем его для одного

слоя сети (чтобы не вводить векторное обозначение для масштабирующих множителей разных слоев).

Обозначим функцию потерь БНС через $L_b(\mathbf{w}, \alpha) = L(\alpha \text{sign}(\mathbf{w}))$. Псевдокод представлен в алгоритме. В предлагаемом методе направление спуска L_b ищется с помощью гладкой функции $\overline{L}_\lambda$, и на этом направлении сравниваются значения функции L_b.

Алгоритм: Обучение БНС с использованием ее непрерывного аналога.

Дано: Инициализированные веса $\mathbf{w}^0 \in \mathbb{R}^n$, функция потерь L, функция потерь БНС L_b, последовательность штрафных коэффициентов $\{\lambda_n\}_{n\in\mathbb{N}}$, параметр метода $\overline{\tau}$, дробитель шага спуска μ, начальный шаг спуска $\gamma_0, \alpha^0 = \alpha^*$ (см. формулу 3).
Вход: номер итерации i, номер эпохи $k, \tau = 0$, (стохастический) градиент $\mathbf{g}_i$ функции $\overline{L}_{\lambda_k}$ в точке $(\mathbf{w}^i, \alpha^i), \gamma = \gamma_0$.
пока *не сходится или не достигнуто максимальное количество итераций* **делать**
  1) Вычисляем $(\mathbf{w}, \alpha) = (\mathbf{w}^i, \alpha^i) - \gamma \mathbf{g}_i$;
  2) Проверяем выполнение $L_b(\mathbf{w}, \alpha) < L_b(\mathbf{w}^i, \alpha^i)$;
  если *условие выполнено* **тогда**
    $\gamma_k = \gamma$, переходим к пункту 3;
  конец
  иначе
    если $\tau < \overline{\tau}$ **тогда**
      Увеличиваем τ на единицу, домножаем γ на μ, возвращаемся к пункту 1;
    конец
    иначе
      принимаем $\gamma = \gamma_0, \tau = 0$, увеличиваем i на единицу (и k, если необходимо) и переходим к моменту входа;
    конец
  конец
  3) $(\mathbf{w}^{i+1}, \alpha^{i+1}) = (\mathbf{w}^i, \alpha^i) - \gamma \mathbf{g}_i$, принимаем $\gamma = \gamma_0, \tau = 0$, увеличиваем i на единицу (и k, если необходимо) и переходим к моменту входа;
конец

4. Эксперименты. Разработанный алгоритм протестирован на задаче классификации выборки CIFAR-10, состоящей из 60000 изображений, разделенных на 10 классов. В качестве базовых архитектур использовались VGG-Small и ResNet-18. У каждой из моделей бинаризовывались веса и активации сверточных слоев, за исключением первого слоя. Масштабирующие множители были инициализированы для каждого канала бинаризуемой свертки по формуле (3) и далее обучены. Размер мини-пакета равнялся 128, начальный шаг спуска $\gamma_0 = 0,1$, дробитель шага спуска $\mu = 0,1, \overline{\tau} = 5$. Для VGG-Small закон изменения λ имел формулу $\frac{1}{n_1}30^{\frac{k}{N}}$, а для ResNet-18 – $\frac{0,01}{n_2}30^{\left(\frac{k}{N}\right)^4}$, где n_1, n_2 – количество бинаризуемых сверточных слоев для VGG-Small и ResNet-18, соответственно, k – номер эпохи, N – количество эпох. Обучение останавливалось, если значение функции потерь приобретало значение nan. Все части программы имплементированы с использованием библиотеки PyTorch.

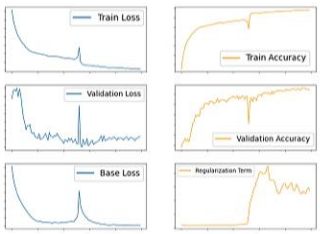


Рис. 2. Графики обучения ResNet-18

На рис. 1 изображен график распределения весов обученной БНС VGG-Small. Видно, что обученные веса имеют бимодальное распределение, что соответствует идее о получении оптимальных бинарных весов. На рис. 2 изображены графики обучения ResNet-18: на горизонтальной оси – номер эпохи, train, validation loss – значения функции потерь L_b на соответствующих выборках, base loss – значение функции потерь L, accuracy – точность БНС на соответствующей выборке, regularization term – величина штрафного слагаемого $\lambda\phi$. Результаты тестирования приведены в таблице. Для сравнения, метод IR-Net [2] на архитектурах VGG-Small и ResNet-18 (на 200 эпохах обучения) достиг точности 90,4 и 91,5, соответственно.

Таблица. Сравнение точностей предложенного алгоритма и IR-Net на CIFAR-10

Метод	**Архитектура**	**Точность (%)**
Алгоритм	VGG-Small	74,3
IR-Net	VGG-Small	90,4
Алгоритм	ResNet-18	76,0
IR-Net	ResNet-18	91,5

5. Заключение. В данной работе предложен новый подход к интерпретации БНС и метод обучения с использованием ее непрерывного аналога. Подробно рассмотрена связь между непрерывными и бинарными сетями, предложена функция потерь с дополнительным штрафным слагаемым, а также разработан алгоритм обучения, обеспечивающий плавный переход к бинарным весам.

Результаты экспериментов на архитектурах VGG-Small и ResNet-18, обученных на датасете CIFAR-10, показали, что предложенный метод позволяет добиться обучения БНС и сформировать веса с бимодальным распределением. Однако, несмотря на значительное уменьшение функции потерь, точность БНС, обученной предложенным методом, пока уступает точности БНС, обученной современными методами, такими как IR-Net. Это оставляет пространство для дальнейших улучшений, включая оптимизацию выбора штрафного множителя λ и адаптацию метода к более сложным архитектурам и задачам.

Литература

1. Yuan C., Agaian S. S. A comprehensive review of binary neural network // Artificial Intelligence Review. 2023. Vol. 56. No 11. P. 12949–13013.

2. Qin H., Gong R. et al. Forward and backward information retention for accurate binary neural networks // Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020. P. 2250–2259.

3. Rastegari M., Ordonez V., Redmon J., Farhadi A. XNOR-Net: Imagenet classification using binary convolutional neural networks // European Conference on Computer Vision (ECCV 2016) / Lecture Notes on Computer Sciences. 2016. Vol. 9908. P. 525–542.

4. Xu S., Li Y. et al. Resilient binary neural network // In Proceedings of the AAAI Conference on Artificial Intelligence. 2023. Vol. 37. No 9. P. 10620–10628.

5. Liu Z., Shen Z. et al. ReActNet: Towards precise binary neural network with generalized activation functions // European Conference on Computer Vision (ECCV 2020) / Lecture Notes on Computer Sciences. 2020. Vol. 12359. P. 143–159.

УДК 519.688

Круглов Н. А.

Программный комплекс для решения задач оптимизации перевозок

Рекомендовано к публикации доцентом Лежниной Е. А.

1. Введение. Данный программный комплекс был разработан для решения задач оптимизации перевозок и предназначен для применения в производственной среде, главным образом, в организациях, работающих в сфере транспортной логистики. Программа нацелена на решение задачи маршрутизации транспорта с временными окнами, ограничениями на грузоподъемность, и другими особенностями, возникшими из конкретного бизнес-кейса. Реализованы также дополнительные модули, упрощающие внедрение программы в бизнес-процессы организации, а также вывод полученных результатов в понятном виде для лица, принимающего решение.

У задач маршрутизации транспорта есть множество вариаций [1]. Также стоит заметить, что эти задачи являются NP-полными, поэтому для их решения используются, как правило, эвристические алгоритмы [2–9]. Существующие реализации многих алгоритмов решения подобных задач сильно отличаются по структуре, в том числе входных и выходных данных, что затрудняет их сравнение и анализ эффективности [10, 11].

В связи с этим была предложена система и единая структура для входных и выходных данных алгоритмов, и при подключении нового алгоритма в приложение необходимо лишь реализовать прослойку, которая будет преобразовывать данные к стандартному для программы формату.

Таким образом архитектура программы позволяет с незначительными доработками адаптировать различные алгоритмы и стратегии решения к задачам транспортной оптимизации, что сокращает время настройки программы под различные бизнес-кейсы и упрощает ее применение в производственной среде.

На данный момент существует множество различных сервисов и библиотек, предназначенных для решения оптимизационных задач, в том числе задач маршрутизации транспорта. Многие из них,

Круглов Николай Аркадьевич – студент, Санкт-Петербургский государственный университет; e-mail: st097166@student.spbu.ru, тел.: +7(988)524-20-55

такие как Яндекс Маршрутизация [12] или Google Routing [13] в бесплатной версии представляют весьма ограниченный функционал, и условия использования не подходят для коммерческого применения.

Существуют и библиотеки с открытым исходным кодом, такие как OR-Tools [14], которая использовалась в первой версии программы. Библиотека рассчитана на достаточно широкий спектр задач. Программе достаточно подать на вход математическую модель задачи, и алгоритм решения построится автоматически, учитывая введенные ограничения. Однако из-за этого алгоритм остается неявным и слабо интерпретируемым, в него сложно вносить изменения и внедрять метрики для оценки текущего решения в процессе работы.

В связи с этим, было решено реализовать новую систему, сохраняющую универсальность, открытость, масштабируемость и доступность для коммерческого использования.

2. Описание программы. Программный комплекс реализован на языке Python и является проектом с открытым исходным кодом. Репозиторий доступен по ссылке [15]. Программа построена как web-сервис с гексагональной архитектурой и состоит из нескольких логических компонентов.

Компонент Entities содержит описание сущностей, которыми оперирует приложение и определяет базовую бизнес-логику системы: order – заказ, address – адрес, client – клиент, deliveryzone – зона доставки, product – товар, vehicle – машина, segment – ребро графа между двумя адресами, route – маршрут, problem – задача оптимизации.

Компонент Ports содержит описания стандартных портов (интерфейсов) взаимодействия сущностей со внешними источниками, обработчиками и потребителями данных: geodatainterface – порт запроса геоданных, routingdatainterface – порт запроса данных о времени пути между точками, businessdatainterface – порт запроса данных о заказах, товарах и автопарке, databaseinterface – порт взаимодействия с локальной базой данных, vrpsolverinterface – порт взаимодействия с алгоритмами оптимизации.

Компонент Adapters содержит адаптеры – реализации портов, преобразующие данные из внешних источников в вид, понятный корневой системе. Таким образом приложение может единообразно взаимодействовать с различными внешними системами, не зная особенностей реализации какой-либо из них.

Компонент Application выступает в качестве посредника между базовой бизнес-логикой и внешними системами через порты и адап-

теры. Содержит логику, описывающую высокоуровневые бизнес-процессы и предоставляет пользовательский интерфейс для лица, принимающего решение, в виде web-приложения.

3. Описание задачи. Требуется по данным о составе, объемах и адресах заказов (узлов), расположении складов а также о характеристиках доступного в автопарке транспорта, составить маршруты для машин, как последовательности узлов и соответственно порядок их загрузки товарами. Для каждой пары узлов задана стоимость пути $f(x_i, x_j)$ из x_i в x_j, которая может быть асимметричной. Необходимо минимизировать суммарную стоимость полученных маршрутов.

Заметим, что, как правило, x_i представляют собой двумерные векторы географических координат (широта и долгота). Однако в исходных бизнес-данных в структуре заказа, как правило, присутствует адрес в текстовом виде, но точных географических координат точки доставки нет. Они могут быть получены из строковых представлений с помощью API геокодинга, например [16]. В свою очередь функция стоимости пути может представлять собой дорожное расстояние между двумя точками и может быть получена с помощью API маршрутизации, такого как [17].

Также в качестве дополнительных условий рассмотрим понятие зоны доставки. Зона доставки – это кластер узлов, расположенных пространственно близко друг к другу. Будем считать, что каждый узел принадлежит одному из кластеров. Из тех или иных соображений считается, что когда машина посещает точку из кластера, далее она должна посетить остальные точки кластера, прежде чем переходить к точкам из другого кластера. Заметим, что алгоритмы, предназначенные изначально для обычной задачи маршрутизации применимы и к такого рода задачам [18, 19].

Также считаем, что склады обслуживают непересекающиеся области. Таким образом, если складов n, то задача разбивается на n подзадач с одним складом, каждая из которых может решаться параллельно и независимо от остальных.

4. Описание стратегии решения с помощью программы. Предлагаемая стратегия решения направлена на снижение размерности исходной задачи за счет ее поэтапного разбиения на множество более мелких подзадач. Каждая из подзадач имеет небольшую размерность и, как правило, может быть эффективно решена даже алгоритмами, имеющими высокую вычислительную сложность.

Рассмотрим задачу для одного склада. В упрощенном виде алгоритм выглядит так:

1. Загрузить конфигурацию адаптеров, используемых для решения.
2. Из BusinessDataInterface запросить данные о заказах, складах и машинах.
3. Проверить наличие географических координат в адресах заказов. Если они отсутствуют, запросить из GeoDataInterface.
4. На множестве упорядоченных пар узлов построить функцию стоимости пути с помощью RoutingDataInterface.
5. Найти центроиды каждой зоны доставки и решить задачу, где узлами являются центроиды.
6. Внутри каждой зоны доставки решить задачу на узлах этой зоны.
7. Для каждой машины субмаршруты, полученные в подзадачах, объединить в целый маршрут.

Таким образом, учитывая нелинейность большинства алгоритмов, время работы алгоритма значительно снижается, что подтверждается вычислительными экспериментами. Сравним среднее время работы одного и того же базового алгоритма целиком на всем множестве точек и при применении его в составе предложенной стратегии. Считаем, что множество точек разделено на 10 зон доставки. Во всех случаях использовался жадный алгоритм (см. таблицу).

Таблица. Время работы алгоритмов

Кол-во узлов	**Среднее время (без кластеров)**	**Среднее время (с кластерами)**
10	0,001	–
100	0,041	0,068
1000	3,901	0,472
10000	391,525	39,79

5. Заключение. Разработанный программный комплекс помогает решать задачи бизнеса, связывая реальные данные с математическими алгоритмами в единой среде, и предоставляя доступный интерфейс для лица, принимающего решение. Дальнейшие улучшения могут включать в себя внедрение новых алгоритмов, построение новых стратегий решения, реализацию методов обработки данных из других источников, а также применение алгоритмов машинного обучения для построения алгоритмов оптимизации, адаптированных под локальные особенности дорожной ситуации.

Литература

1. Vidal T., Laporte G., Matl P. A concise guide to existing and emerging vehicle routing problem variants // European Journal of Operational Research. 2019. No 286. P. 401–416.
2. Lin S., Kernighan B. W. An effective heuristic algorithm for the traveling-salesman problem // Operations Research. 1973. No 21. P. 498–516.
3. Helsgaun K. An effective implementation of the Lin–Kernighan traveling salesman heuristic // European Journal of Operational Research. 2000. No 126. P. 106–130.
4. Hitoshi K., Junichi O. Solving time-dependent traveling salesman problems using ant colony optimization based on predicted traffic // Distributed Computing and Artificial Intelligence. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012. P. 25–32.
5. Eneko O., Xin-She Y., Javier D. S. Traveling salesman problem: a perspective review of recent research and new results with bio-inspired metaheuristics // Nature-Inspired Computation and Swarm Intelligence. Academic Press, 2020. P. 401–416.
6. Эвристики в задаче коммивояжера // Synset [Электронный ресурс]: URL:https://www.synset.com/ai/ru/tsp/Salesman_Heuristics.html (дата обращения: 19.03.2025).
7. Бушуева А. С., Власова Т. В. Применение эвристических алгоритмов для решения задачи маршрутизации с несколькими складами // Процессы управления и устойчивость. 2024. Т. 11. No 1. С. 307–311.
8. Маркелова А. Ю., Аллахвердян А. Л., Мартемьянов А. А. и др. Прикладная задача маршрутизации для парка беспилотных летательных аппаратов с использованием модифицированного параллельного генетического алгоритма // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2022. Т. 18. No 1. С. 135–148.
9. Захаров А. О., Коваленко Ю. В. Построение и сужение множества Парето в асимметричной задаче коммивояжера с двумя критериями // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2018. Т. 14. No 4. С. 378–392.

10. Lin-Kernighan heuristic in Python [Электронный ресурс]: URL: https://github.com/kikocastroneto/lk_heuristic (дата обращения: 19.03.2025).
11. VRP solution with ant colony optimization [Электронный ресурс]: URL:https://github.com/Smmehdihosseini/AntCVRP (дата обращения: 19.03.2025).
12. Программа для составления маршрута // Яндекс Маршрутизация [Электронный ресурс]: URL:https://yandex.ru/routing/route-planner/ (дата обращения: 19.03.2025).
13. Route optimization API // Google Maps Platform [Электронный ресурс]: URL:https://developers.google.com/maps/documentation/route-optimization (дата обращения: 19.03.2025).
14. OR-Tools // Google for Developers [Электронный ресурс]: URL: https://developers.google.com/optimization (дата обращения: 19.03.2025).
15. Logistics optimization program [Электронный ресурс]: URL:https://github.com/Kapitan-Avaria/Logistics-Optimization-Program (дата обращения: 19.03.2025).
16. Geocoder API // Yandex Maps API [Электронный ресурс]: URL:https://yandex.com/maps-api/docs/geocoder-api/index.html (дата обращения: 19.03.2025).
17. OpenRouteService [Электронный ресурс]: URL:https://openrouteservice.org/ (дата обращения: 19.03.2025).
18. Helsgaun K. Solving the clustered traveling salesman problem using the Lin-Kernighan-Helsgaun algorithm [Электронный ресурс]: URL:https://www.researchgate.net/publication/266382168_Solving_the_Clustered_Traveling_Salesman_Problem_Using_the_Lin-Kernighan-Helsgaun_Algorithm (дата обращения: 19.03.2025).
19. Yongliang L., Jin-Kao H., Qinghua W. Solving the clustered traveling salesman problem via TSP methods // arXiv preprint arXiv:2007.05254v3, 2022.

УДК 517.977

Кузнецова Д. С.

Исследование методов инициализации алгоритма роя частиц (PSO) для решения задач глобальной оптимизации

Рекомендовано к публикации профессором Крылатовым А. Ю.

1. Введение. В современной науке наблюдается увеличение количества исследований, посвященных методам оптимизации на основе эвристических и метаэвристических алгоритмов. Это объясняется тем, что многие задачи являются NP-трудными и нахождение точного решения на больших размерностях становится невозможным. Усложняется постановка задачи и тем, что у целевой функции могут отсутствовать свойства выпуклости, дифференцируемости, унимодальности [1].

В этом случае используют приближенные методы оптимизации, одним из которых является алгоритм роя частиц (PSO) [2] и его модификации. В различных областях знаний, от логистики до медицины, есть исследования, посвященные этому подходу. Особый интерес представляет его использование в решении двухуровневых задач оптимизации, например, в задачах равновесного распределения транспортных потоков [3, 4].

На текущий момент одним из актуальных направлений исследований является изучение различных вариантов адаптаций данного алгоритма [5], которые повышают эффективность оптимизации. Основная проблема, с которой сталкивается PSO – это преждевременная конвергенция частиц, в результате чего агенты популяции остаются в локальном оптимуме [6]. Это может быть вызвано недостаточным разнообразием роя [7]. Для решения такой проблемы предлагается модифицированная версия алгоритма PSO для поиска глобального минимума на основе различных методов инициализации начальных позиций агентов, используя псевдослучайные и квазислучайные последовательности.

Кузнецова Дарья Сергеевна – аспирант, Санкт-Петербургский государственный университет; e-mail: daria.kuznetsova@spbu.ru, тел.: +7(918)786-83-83

2. Цель работы. Исследование влияния способов инициализации роя в алгоритме PSO на эффективность оптимизации, реализация алгоритма и его модификаций на языке Python и его проверка на различных тестовых функциях.

3. Описание алгоритма роя частиц. Алгоритм роя частиц был предложен в 1995 году социальным психологом Джеймсом Кеннеди и инженером-электриком Расселом Эмберхартом [8]. Алгоритм использовался для оптимизации непрерывных нелинейных функций.

На первом этапе каждая частица инициализируется либо как случайное решение на области допустимых значений оптимизируемой функции, либо в соответствии с заданным правилом. Далее ей присваивается вектор скорости — значение, на которое следует переместить частицу по области поиска. Это значение обновляется на всех итерациях. Для частицы рассчитывается значение целевой функции. Частица запоминает лучшее решение, которое она нашла и рой за все пройденные итерации.

Обновление вектора скорости реализуется по формуле

$$v_{ij}(t+1) = v_{ij}(t) + c_1 r_1 (p_{ij}(t) - x_{ij}(t)) + c_2 r_2 (p_{gj}(t) - x_{ij}(t)),$$

где i — количество частиц в рое, $i \in [1, \ldots, n]$; j — количество переменных в исследуемой функции, $j \in [1, \ldots, d]$; v_{ij} — компонента скорости частицы i на итерации t; $V_i = [v_{i1}, \ldots, v_{id}]$ — вектор скорости частицы i, где $v_{ij} \in [v_{\min}, v_{\max}]$; x_{ij} — координата частицы i на итерации t алгоритма; p_{ij} — координата лучшего решения, найденного частицей; $P_i = [p_{i1}, \ldots, p_{id}]$ — вектор координат, соответствующий лучшему найденному решению частицы i на предыдущих итерациях; p_{gj} — координата j лучшего решения, найденного всеми частицами; $P_g = [p_{g1}, \ldots, p_{gd}]$ — вектор координат, соответствующий лучшему решению, которое было найдено частицами на предыдущих итерациях; r_1, r_2 — случайные числа на интервале $(0, 1)$; c_1, c_2 — весовые коэффициенты.

Далее корректируем текущую координату каждой частицы

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1).$$

После рассчитывается значение целевой функции в каждой новой точке, частица проверяет, не стала ли новая координата лучшей среди всех точек, где она побывала. Затем среди всех новых точек

проверяем, не нашли ли мы новую глобально лучшую точку. В случае если нашли, то запоминаем ее координаты и значение целевой функции в ней.

4. Способы инициализации роя. Выбор подхода к инициализации роя является важным этапом при разработке алгоритма PSO. Предполагается, что эффективность работы PSO сильно зависит от начального расположения роя, поскольку оно определяет последующее движение алгоритма по области допустимых значений. В некоторых случаях может возникнуть неравномерное распределение частиц и их слишком большое количество на определенных участках целевой функции при недостаточном количестве на других, особенно в задачах с большой размерностью. В работе сравниваются три метода генерации частиц.

1. Случайная инициализация роя с равномерным распределением.
2. Использование генератора квазислучайной последовательности с низким расхождением.
3. Использование генератора псевдослучайных чисел.

4.1. Случайная инициализация роя с равномерным распределением. В данном случае каждая частица роя представляет собой вектор, каждая координата которого выбирается равновероятно из отрезка некоторых допустимых для нее значений $[x_{\min}, x_{\max}]$:

$$x_i(0) \sim U(x_{i,\min}, x_{i,\max}).$$

4.2. Квазислучайные последовательности с низким расхождением для инициализации роя. Квазислучайная последовательность с низким расхождением — это метод построения последовательных точек детерминированным образом, который уменьшает вероятность скапливания (расхождения), в то же время обеспечивая равномерное покрытие всего пространства («синий шум»).

В работе рассматривается два способа построения последовательностей по методу построения их базисных параметров [9]:

- квазислучайная последовательность Кронекера на основе иррациональных дробей. Каноническая рекурсивная последовательность Кронекера определяется, как

$$R_1(\alpha) : t_n = (s_0 + n\alpha)^{-1}; \quad n = 1, 2, \ldots,$$

где α – любое иррациональное число. В работе параметры генератора были определены как $s_0 = 0$ и $\alpha = \frac{\sqrt{5}-1}{2}$ – обратное золотое сечение.

- квазислучайная последовательность Хэлтона на основе простых чисел. Последовательность чисел Хэлтона создана последовательностями Ван дер Корпута и конструируется на основании радикальной инверсии, которая может быть выражена однозначно для основания b через последовательность $d_n(t), \ldots, d_1(t)$ по формуле

$$t = \sum_{i=1}^{n} d_i(t) b^{i-1},$$

где все числа $d_i(t) \in (0, b-1)$.

4.3. Последовательность псевдослучайных чисел для генерации роя. Псевдослучайная последовательность (ПСП) – последовательность чисел, которая была вычислена по некоторому определенному арифметическому правилу, но имеет все свойства случайной последовательности чисел в рамках решаемой задачи.

- Линейный конгруэнтный ГПСЧ. Метод заключается в вычислении членов последовательности, определяемых рекуррентной формулой:

$$x_{i+1} = (\alpha x_i + c) \bmod m,$$

где α, c, m – некоторые целочисленные коэффициенты.

Получаемая последовательность зависит от выбора стартового числа x_0 и при разных его значениях получаются различные последовательности случайных чисел.

- Мультипликативный конгруэнтный ГПСЧ. Основная формула метода имеет вид

$$x_{i+1} = \alpha x_i \bmod m,$$

где α и m – неотрицательные целые числа. Согласно этому выражению, требуется взять последнее случайное число x_i, умножить его на постоянный коэффициент α и взять остаток от деления на m. Поэтому для генерирования последовательности чисел x_i необходимо задать начальное значение x_0, множитель α и делитель m.

5. Анализ эффективности методов инициализации роя. Для оценки предложенных методов применяются тестовые функции с известными глобальными минимумами. Результаты экспериментов демонстрируют, как различные стратегии инициализации влияют на работу алгоритма на функциях различного вида.

5.1. Тестовые функции. Для проверки эффективности разработанных модификаций алгоритма использовались унимодальные и мультимодальные тестовые функции, представленные в таблице 1.

Таблица 1. Описание тестовых функций

f	**Название функции**	**Функция**	**Глобальный минимум**
1. Унимодальные тестовые функции			
f_1	Функция Бута	$(x+2y-7)^2+(2x+y-5)^2$	0.0
f_2	Функция Матьяса	$0{,}26(x^2+y^2)-0{,}48xy$	0.0
f_3	Функция Изома	$-\cos(x)\cos(y)\exp(-((x-\pi)^2+(y-\pi)^2))$	-1.0
2. Мультимодальные тестовые функции			
f_4	Функция Экли	$-20\exp[-0{,}2\sqrt{0{,}5(x^2+y^2)}-$ $\exp[0{,}5(\cos(2\pi x)+\cos(2\pi y))]+e+20]$	0.0
f_5	Функция Химмельблау	$(x^2+y-11)^2+(x+y^2-7)^2$	0.0
f_6	Функция Хольдера	$-\lvert\sin x\cos y\exp(\lvert 1-\frac{\sqrt{x^2+y^2}}{\pi}\rvert)\rvert$	-19.2085

5.2. Результаты экспериментов. Классический алгоритм роя частиц со случайной инициализацией роя (RPSO) дополнялся рядом модификаций. Алгоритм и его модификации были реализованы на языке Python [11]. Были использованы квазислучайные последовательности Хэлтона (HPSO) и Кронекера (KPSO), псевдослучайные последовательности на основе линейного конгруэнтного метода (LCPSO) и мультипликативного конгруэнтного метода (MCPSO) для инициализации начальных частиц. В таблице 2 собраны усредненные значения лучшего найденного роем значения целевой функции для запусков алгоритма со следующими характеристиками:

- количество запусков каждого алгоритма – 100;
- количество итераций на каждом запуске $T = 100$;
- количество частиц в рое i – 50; 250; 1250;
- весовые коэффициенты $c_1 = c_2 = 0{,}5$;
- коэффициент инерции $w = 0{,}8$;
- $r_1 = r_2$ – случайные числа на интервале $(0, 1)$.

Таблица 2. Результаты применения различных техник инициализации роя

f	i	**RPSO**	**KPSO**	**HPSO**	**LCPSO**	**MCPSO**
1. Унимодальные тестовые функции						
f_1	50	$5{,}22{\cdot}10^{-12}$	$4{,}25{\cdot}10^{-12}$	$4{,}17{\cdot}10^{-12}$	$5{,}45{\cdot}10^{-12}$	$4{,}07{\cdot}10^{-12}$
	250	$2{,}45{\cdot}10^{-12}$	$1{,}85{\cdot}10^{-14}$	$1{,}45{\cdot}10^{-13}$	$2{,}16{\cdot}10^{-12}$	$3{,}21{\cdot}10^{-14}$
	1250	$1{,}6{\cdot}10^{-13}$	$1{,}45{\cdot}10^{-15}$	$1{,}11{\cdot}10^{-13}$	$1{,}8{\cdot}10^{-13}$	$2{,}65{\cdot}10^{-15}$
f_2	50	$1{,}3{\cdot}10^{-14}$	$5{,}15{\cdot}10^{-17}$	$3{,}81{\cdot}10^{-21}$	$1{,}15{\cdot}10^{-12}$	$2{,}66{\cdot}10^{-16}$
	250	$3{,}41{\cdot}10^{-14}$	$3{,}81{\cdot}10^{-18}$	$4{,}83{\cdot}10^{-23}$	$1{,}71{\cdot}10^{-11}$	$2{,}14{\cdot}10^{-24}$
	1250	$5{,}41{\cdot}10^{-16}$	$3{,}25{\cdot}10^{-18}$	$5{,}15{\cdot}10^{-22}$	$1{,}85{\cdot}10^{-11}$	$6{,}45{\cdot}10^{-25}$
f_3	50	$-0{,}99$	$-0{,}99$	$-0{,}99$	$-0{,}99$	$-0{,}99$
	250	$-0{,}99$	$-0{,}99$	$-0{,}99$	$-0{,}99$	$-0{,}99$
	1250	$-0{,}99$	$-0{,}99$	$-0{,}99$	$-0{,}99$	$-0{,}99$
2. Мультимодальные тестовые функции						
f_4	50	$8{,}79{\cdot}10^{-06}$	$5{,}24{\cdot}10^{-15}$	$6{,}18{\cdot}10^{-19}$	$7{,}58{\cdot}10^{-17}$	$5{,}75{\cdot}10^{-21}$
	250	$8{,}60{\cdot}10^{-07}$	$7{,}17{\cdot}10^{-15}$	$3{,}18{\cdot}10^{-12}$	$6{,}25{\cdot}10^{-17}$	$3{,}24{\cdot}10^{-11}$
	1250	$2{,}23{\cdot}10^{-06}$	$6{,}18{\cdot}10^{-16}$	$4{,}21{\cdot}10^{-12}$	$6{,}16{\cdot}10^{-14}$	$2{,}61{\cdot}10^{-12}$
f_5	50	$6{,}19{\cdot}10^{-12}$	$3{,}24{\cdot}10^{-45}$	$3{,}18{\cdot}10^{-39}$	$4{,}15{\cdot}10^{-11}$	$3{,}66{\cdot}10^{-25}$
	250	$3{,}63{\cdot}10^{-11}$	$7{,}75{\cdot}10^{-79}$	$3{,}85{\cdot}10^{-41}$	$5{,}17{\cdot}10^{-12}$	$2{,}65{\cdot}10^{-24}$
	1250	$2{,}79{\cdot}10^{-12}$	$6{,}54{\cdot}10^{-75}$	$4{,}45{\cdot}10^{-59}$	$5{,}01{\cdot}10^{-12}$	$2{,}15{\cdot}10^{-24}$
f_6	50	$-19{,}2$	$-19{,}2$	$-19{,}2$	$-19{,}2$	$-19{,}2$
	250	$-19{,}2$	$-19{,}2$	$-19{,}2$	$-19{,}2$	$-19{,}2$
	1250	$-19{,}2$	$-19{,}2$	$-19{,}2$	$-19{,}2$	$-19{,}2$

Как видно из результатов, представленных в таблице 2, большинство функций KPSO, HPSO и MCPSO имеют почти одинаковый результат. Эти три метода имеют одинаковую производительность для решения проблемы сходимости и достигают глобальных оптимумов. При этом можно заметить, что для мультимодальных тестовых функций алгоритмы, использующие эти подходы к инициализации начальных точек работают более эффективно, чем их случайная реализация (RPSO). Это можно объяснить тем, что использование псевдослучайных и квазислучайных последовательностей гарантирует более равномерное распределение начальных решений целевой функции по области допустимых значений.

6. Заключение. Таким образом, несмотря на то, что алгоритм роя частиц может использоваться для решения различных задач, требуется тщательное исследование его возможных адаптаций для более эффективной оптимизации. В ходе исследования было предложено несколько модификаций алгоритма (RPSO, KPSO, HPSO, LCPSO, MCPSO) на основе изменения подхода к инициализации начального роя, в том числе с использованием квазислучайных и псевдослучайных последовательностей. В работе представлено подробное описание предложенных методов и результаты их тестирования на различных функциях. Предложенные подходы могут помочь в решении проблемы локальных минимумов, которая приводит к преждевременной сходимости, для получения более точных результатов.

Литература

1. Карпенко А. П. Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой: учебное пособие. М.: Изд-во МГТУ им. Н. Э. Баумана, 2017. 446 с.
2. Alaia B., Harbaoui I., Borne P., Bouchriha H. A comparative study of the PSO and GA for the m-MDPDPTW // International Journal of Computers Communications & Control. 2018. Vol. 13. No 1. P. 8–23.
3. Крылатов А. Ю. Оптимальные стратегии управления транспортными потоками на сети из параллельных каналов // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2014. Т. 10. № 2. С. 121–130.

4. Крылатов А. Ю. Оптимальные стратегии управления пропускной способностью маршрутов транспортной сети // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2017. Т. 13. № 2. С. 182–192.

5. Васильев В. А., Кузнецова Д. С. Алгоритм роя частиц и анализ его эффективности для решения задач безусловной однокритериальной оптимизации // Процессы управления и устойчивость. 2022. Т. 9. № 1. С. 190–197.

6. Yang C., Gao W., Liu N., Song C. Low-discrepancy sequence initialized particle swarm optimization algorithm with high-order nonlinear time-varying inertia weight // Applied Soft Computing. 2015. Vol. 29. P. 386–394.

7. Chen B., Lei H., Shen H., Liu Y., Lu Y. A hybrid quantum-based PIO algorithm for global numerical optimization // Science China Information Sciences. 2019. Vol. 62. No 7. P. 1–12.

8. Kennedy J., Eberhart R. Particle swarm optimization // Proceedings of International Conference of Neural Networks. 1995. P. 1942–1948.

9. Weerasinghe G., Cao Y. Particle swarm optimization simulation via optimal Halton sequences // Procedia Computer Science. 2016. Vol. 80. P. 772–781.

10. Francois M., Grosges T., Barchiesi D., Erra R. Pseudo-random number generator based on mixing of three chaotic maps // Communications in Nonlinear Science and Numerical Simulation. 2016. Vol. 80. P. 772–781.

11. Кузнецова Д. С. PSO-algorithm-CPS-25- [Электронный ресурс]: URL: `https://github.com/daturare/PSO-algorithm-CPS-25-` (дата обращения: 20.03.2025).

УДК 004.021

Кукло Р. А.

Исследование методов сбора, обработки и сжатия данных с помощью граничных вычислений в AIoT

Рекомендовано к публикации доцентом Корховым В. В.

1. Введение. Artificial Intelligence of Things (AIoT) – это система подключенных устройств, способных получать, обрабатывать и передавать данные по сети, при этом модели искусственного интеллекта (ИИ) обучаются на полученных данных и имеют возможность управлять устройствами системы. Например, это может быть система мобильных пульсометров, отслеживающих частоту сердечных сокращений людей. Данные с устройств передаются в облако и обрабатываются нейронной сетью. Результатом работы сети может быть вероятность возникновения сердечных заболеваний. Данные результаты и рекомендации системы могут быть отражены в приложении у пользователя.

Современные AIoT системы все чаще используют граничные вычисления для анализа данных в непосредственной близости к сенсорам. Устройства, находящиеся в граничных слоях, обладают меньшими вычислительными мощностями по сравнению с облачными вычислениями, однако данный подход приводит к более эффективному использованию ресурсов сети и уменьшению латентности [1].

С ростом количества устройств возникают новые проблемы, такие как хранение и передача большого объема данных, сложность в интегрировании ИИ в IoT, конфиденциальности данных, высокая стоимость технологий и затраты на внедрения, увеличение времени на предоставление результатов анализа. В данном исследовании рассмотрена проблема эффективного сбора и сжатия данных.

2. Постановка задачи. Пусть имеются n сенсоров и каждый собирает данные, которые хранятся в виде временных рядов. Каждому сенсору на плоскости присвоены координаты (x, y).

Кукло Роман Алексеевич – студент, Санкт-Петербургский государственный университет; e-mail: st097551@student.spbu.ru, тел.: +7(921)031-90-99

Задача состоит в анализе методов сжатия временных рядов для более продуктивной работы AIoT.

Для начала будут рассмотрены данные, затем – методы сжатия временных рядов.

3. Данные. Были использованы реальные данные с сенсоров. Из открытых датасетов выбран NAB – данные о дорожном движении, собранные Департаментом транспорта Миннесоты [2]. Для алгоритма сжатия взяты данные о скорости потока автомобилей (целые числа).

Алгоритм Sprintz [3] требует данные с нескольких сенсоров, поэтому временной ряд (2500 значений) разделен на 5 рядов по 500 значений: в первый ряд включен каждый 5-й элемент, начиная с первого, во второй – со второго и т. д. Полученные ряды объединены в датафрейм, где столбцы соответствуют сенсорам.

Для алгоритмов XOR [4] и LZ4 [5] использовался Emhires Dataset [6]. Он представляет данные 23 ветряных установок со значениями, равными отношению мощности к максимальной мощности на соответствующей установке. Для тестирования были взяты шесть сенсоров и их первые 10000 значений. Данные из датасета крайне разреженные из-за отсутствия непрерывности в их замерах. Поэтому после удаления нулевых строк, не несущих информации, осталось 5371 строка. Для полученных данных можно вычислить матрицу корреляции (см. рис. 1).

	sensor_0	sensor_1	sensor_2	sensor_3	sensor_4	sensor_5
sensor_0	1.000000	0.692844	0.718031	0.858738	0.901123	0.844601
sensor_1	0.692844	1.000000	0.533855	0.723919	0.705373	0.862921
sensor_2	0.718031	0.533855	1.000000	0.637662	0.685302	0.661292
sensor_3	0.858738	0.723919	0.637662	1.000000	0.772007	0.839008
sensor_4	0.901123	0.705373	0.685302	0.772007	1.000000	0.884198
sensor_5	0.844601	0.862921	0.661292	0.839008	0.884198	1.000000

Рис. 1. Матрица корреляции для тестовых данных Emhires Dataset

4. Алгоритмы сжатия. Сжатие данных при передаче экономит ресурсы сети, но в AIoT важно сохранять качество информации. Большинство методов сжатия временных рядов работают с потерями, включая аппроксимацию полиномами низкого порядка [7]. Еще одним методом является сжатие на основе нейронных сетей. Однако

для данного метода требуется больше времени для предобработки данных и обучения.

Рассмотрим алгоритмы без потерь. Один из них – XOR-сжатие чисел с плавающей точкой, применяемое в Time Series Database Gorilla [4]. В оригинальном исследовании этот метод уменьшил объем данных более чем в 10 раз, но эффективность связана с высокой частотой добавления значений, что делает ряды более гладкими.

В основе этого алгоритма лежит подсчет XOR функции между последовательными значениями ряда. Однако на экспериментальных данных алгоритм показал несильное сжатие. Весь объем в 258 килобайт уменьшился в 1,03 раз. Отдельно отметим, что данный алгоритм плохо справляется с резкими перепадами в значениях рядов. Если в какой-то момент произошел спад, а затем данные начали изменяться плавно, то алгоритм будет практически не сжимать плавно изменяющиеся данные до следующего резкого изменения.

Еще одним популярным алгоритмом сжатия является алгоритм Sprintz. Это наиболее эффективный алгоритм сжатия целочисленных данных временных рядов целых 8-битных или 16-битных значений. Существует несколько вариаций этого алгоритма. Рассмотрим одну из них, наиболее быструю. Sprintz – это по сути суперпозиция нескольких алгоритмов кодирования сразу нескольких временных рядов одновременно [3].

На вход алгоритму подается датафрейм, где каждой колонке соответствует временной ряд. Затем этот датафрейм разбивается на батчи, в оригинальной статье размер батча равен 8. Первая строка каждого батча остается без изменений. К остальным применяется delta-кодирование и ZigZag-кодирование. После весь батч можно представить в виде битовой строки, содержащей заголовок и значения ZigZag кодирования. В результате весь объем целочисленных данных уменьшился в 1,13 раз. Данный алгоритм плохо сжимает разреженные данные, а также зависит от размера батча. К примеру, наилучшего сжатия для тестовых данных удалось достичь при размере батча равным 12.

Заключительным алгоритмом является LZ4 [5]. В исследовании использовалась библиотека lz4. Алгоритм сжимает данные, заменяя повторяющиеся фрагменты ссылками на уже встречавшиеся участки. Каждая запись состоит из литеральной части, обозначающей уникальные данные, записываемые без изменений, и ссылки на совпадения, т. е. данные, которые уже встречались. Если же совпаде-

ние слишком длинное, то алгоритм использует дополнительные байты для его указания. LZ4 не применяет сложные методы кодирования, поэтому сжатие и распаковка происходят очень быстро. И чем дольше происходит поиск совпадающих паттернов, тем быстрее происходит декомпрессия. LZ4 уменьшил размер данных с плаваюшей точкой в 1,34 раз.

Теперь перейдем к сжатию с потерями. Одним из наилучших алгоритмов сжатия с потерями для временных рядов является алгоритм SZ3. Ключевой особенностью SZ3 является использование простых предикторов, таких как Lorenzo Predictor и Linear Interpolation Predictor. После предиктора вычисляется ошибка предсказания и данные ошибки квантуются. На этом этапе и теряются данные. Затем используется кодирование по Хаффману. Выбор наилучшего предиктора для каждой временной точки позволяет уменьшить ошибку декодирования. Указанный алгоритм достаточно эффективен.

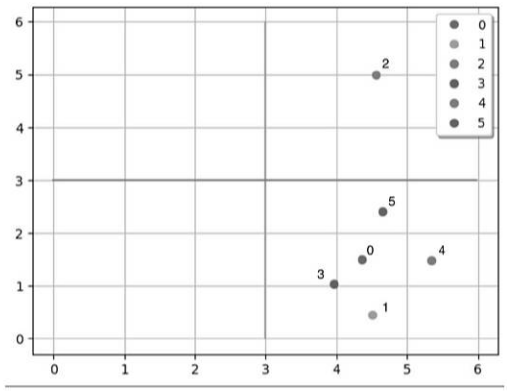


Рис. 2. Пример расположения сенсоров на плоскости

5. Обход сенсоров и PCA. Присвоим каждому сенсору координаты (x, y), исходя из того факта, что сенсоры с коррелирующими значениями находятся близко (см. рис. 2). В противном случае, данная задача не имела бы смысла.

Применим алгоритм обхода сенсоров, описанный в [8]. При последовательном обходе сенсоров будут формироваться кластеры, где временные ряды сильно коррелируют. Применим алгоритмы XOR и

Sprintz для сжатия данных кластеров. При чем на вход каждого алгоритма будут поступать данные главного сенсора, а вместо зависимых – их разность с главным. Данный алгоритм оказался неэффективным для Sprintz. Однако XOR позволил сжать данные в 1,49 раз.

Рассмотрим использование алгоритма LZ4 при обходе сенсоров. В каждом кластере чем сильнее коррелируют данные, тем сильнее их можно сжать с помощью метода главных компонент (PCA) [9]. К этому результату можно применить LZ4. Алгоритм при таком обходе будет следующим:

1. Начинается обход сенсоров и формирование кластера.
2. Если в кластере количество сенсоров $n > 1$:
   а) используем PCA для сжатия. Берем число главных компонент как $\max(2, \lfloor 0{,}5n \rfloor)$;
   б) сжимаем результаты, компоненты PCA, средние значения и дисперсии с помощью LZ4.
3. Иначе, сжимаем ряд с помощью LZ4.

Таким образом, применение данного алгоритма позволило сжать данные в 1,73 раз. При восстановлении данных, среднеквадратичная ошибка равна 0,001. Стоит отметить, что данный метод относится к сжатию с потерями из-за применения метода главных компонент, однако ошибка при восстановление данных будет только для тех сенсоров, которые были объединены в кластер.

6. Обход сенсоров и CNN. Теперь же будем максимально учитывать факт корреляции зависимых сенсоров с главным. Логично передавать в полной мере информацию только о главном сенсоре, а зависимые пытаться восстановить по нему. Примем во внимание, что для работы системы в режиме реального времени и ограничениях памяти на самих сенсорах необходимо часто передавать небольшой объем данных. Поэтому рассмотрим неглубокую сверточную нейронную сеть (CNN), способную учитывать временные зависимости, не требующую больших вычислительных затрат на обучение и предсказание результатов. Кроме того, легкость архитектуры должна обладать способностью отличного сжатия.

Рассмотрим архитектуру такой модели. Входные данные поступают в первый слой – сверточный (Separable Conv1D) с размером ядра 3 и двумя выходными каналами. Получившиеся данные поступают в два последовательных полносвязных слоя. Первый скрытый слой имеет 4 нейрона и функцию активации SELU. Второй выходной слой имеет столько нейронов, сколько в данном кластере зависимых сенсоров, и функцию активации Sigmoid. Это связано с природой исходных данных. Для обучения используется оптимизатор Adam, а само обучение занимает 50 эпох. В качестве функции потерь имеет смысл использовать MSE (Mean Squared Error).

Алгоритм	Данные	Коэффициент сжатия	MSE
Sprintz	NAB	1,13	-
XOR	Emhires Dataset	1,03	-
LZ4	Emhires Dataset	1,34	-
SZ3 (together)	Emhires Dataset	22,3	0,00026
SZ3 (separately)	Emhires Dataset	20,4	0,0003
Sprintz + clustering	NAB	0,68	-
XOR + clustering	Emhires Dataset	1,49	-
PCA + clustering	Emhires Dataset	1,73	0,001
CNN + clustering	Emhires Dataset	21,76	0,004

Рис. 3. Сравнительная характеристика результатов алгоритмов сжатия

Для сжатия параметров модели без потерь был использован алгоритм LZ4. Высокая скорость LZ4 позволяет быстро сжать модель без потерь, а также быстро декодировать после передачи в граничные слои. Для сжатия данных главного сенсора и сенсоров, не вошедших в кластеры, можно использовать алгоритм SZ3 с наименьшей допустимой погрешностью декодирования. Данный метод позволяет конкурировать по коэффициенту сжатия с алгоритмом SZ3.

7. Заключение. Таким образом, в данной работе рассмотрены различные методы сжатия временных рядов. XOR-сжатие вместе с кластеризацией позволило лучше всего сжать данные без потерь почти в 1,5 раз. А комбинация обхода сенсоров с использованием CNN дала наилучший коэффициент сжатия в 21,76 раз (см. рис. 3).

Литература

1. Chen J., Ran X. Deep learning with edge computing: a review // Proceedings of the IEEE. 2019. No 8. P. 1655–1674.
2. NAB/data [Электронный ресурс]: URL:https://github.com/numenta/NAB/tree/master/data/realTraffic (дата обращения: 15.11.2024).
3. Blalock D., Madden S., Guttag J. Sprintz: Time series compression for the internet of things // Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies. 2018. No 3. P. 1–23.
4. Pelkonen T. Gorilla: A fast, scalable, in-memory time series database // Proceedings of the VLDB Endowment. 2015. No 12. P. 1816–1827.
5. Bartik M., Ubik S., Kubalik P. LZ4 compression algorithm on FPGA // IEEE International Conference on Electronics, Circuits, and Systems (ICECS). 2015. P. 179–182.
6. Wind generation [Электронный ресурс]: URL:https://www.kaggle.com/datasets/sohier/30-years-of-european-wind-generation?resource=download (дата обращения: 20.12.2024).
7. Keogh E. Dimensionality reduction for fast similarity search in large time series databases // Knowledge and information Systems. 2001. P. 263–286.
8. Кукло Р. А. Применение метода кластеризации по времени и географическим данным для уменьшения объема передаваемых данных // Процессы управления и устойчивость. 2024. Т. 11. № 1. С. 212–217.
9. Mackiewicz A., Ratajczak W. Principal components analysis (PCA) // Computers and Geosciences. 1993. No 3. P. 303–342.

УДК 519.642

Лаврухин М. Ю.

Применение генетического алгоритма для оптимизации трёхмерных траекторий на рельефе

Рекомендовано к публикации профессором Аббасовым Э. М.

1. Введение. Рассматривается задача построения оптимального маршрута в трёхмерном пространстве (x, y, z), где высота z задаётся функцией $\varphi(x, y)$. Стоимость прокладки такого маршрута учитывает как геометрическую длину, так и различные дополнительные факторы (логистика доставки материалов, сложность грунта и т. д.). Задача поиска оптимальных траекторий на сложном рельефе встречается в дорожном строительстве, геодезии, робототехнике и многих других областях прикладной математики и вычислительной механики, и её можно формализовать через вариационный функционал, совмещающий длину пути и дополнительные факторы стоимости.

Для поиска решения, близкого к глобальному оптимуму, предлагается применять генетический алгоритм (ГА). При этом выполнен переход от двумерного случая, где z не учитывается, к трёхмерному вариационному функционалу, который подвергся дискретизации. В работе [1] генетический алгоритм успешно применялся для построения кратчайшего маршрута с нелокальными условиями, что позволило оценить его эффективность. В работе [2] было предложено общее описание вариационного функционала и приведены некоторые точные решения на базе динамического программирования.

В настоящей статье используется и уточняется тот же интеграл стоимости, минимизируемый при помощи *эволюционного* метода. Представлены результаты вычислительных экспериментов, в которых найденные ГА-траектории сравниваются со ссылочным (полуаналитическим) решением, а также обсуждаются перспективы дальнейшего развития метода. Генетические алгоритмы, являющиеся од-

Лаврухин Михаил Юрьевич – магистрант, Санкт-Петербургский государственный университет; e-mail: st076244@student.spbu.ru, тел.: +79193156601

ним из наиболее известных подходов эволюционных вычислений [3], успешно применяются для многокритериальной оптимизации маршрутов и траекторий [4].

2. Постановка задачи. Рассмотрим прямоугольную область $\{\,0 \leqslant x \leqslant l,\ 0 \leqslant y \leqslant y_l\}$. Каждая точка (x, y) однозначно отображается на высоту $z = \varphi(x, y)$. По аналогии с [2] пусть $\alpha(x, y)$ отражает удорожание строительства дороги (например, затраты на доставку материалов), а $\beta(x, y)$ учитывает местные условия (структуру почвы, экологию, грунтовую сложность и т. д.). Тогда стоимость всей траектории Q определяется вариационным функционалом вида

$$\begin{aligned} J(y) \;=\; & \int_0^l \alpha\big(x, y(x)\big)\,\sqrt{1 + \big(y'(x)\big)^2 + \big(z'(x)\big)^2}\times \\ & \times \int_0^x \sqrt{1 + \big(y'(\xi)\big)^2 + \big(z'(\xi)\big)^2}\,d\xi\,dx + \\ & + \int_0^l \beta\big(x, y(x)\big)\,\sqrt{1 + \big(y'(x)\big)^2 + \big(z'(x)\big)^2}\,dx, \end{aligned} \tag{1}$$

где

$$z'(x) \;=\; \frac{\partial\varphi}{\partial x}\big(x, y(x)\big) \;+\; \frac{\partial\varphi}{\partial y}\big(x, y(x)\big)y'(x).$$

Граничные условия $y(0) = 0$, $y(l) = y_l$. Как показано в [2], функционал не сводится к простой задаче кратчайшего пути – присутствует «самоинтегрирующаяся» компонента с α, зависящая от уже пройденного расстояния.

2.1. Дискретизация функционала. Для численного решения используем следующую схему:

- строим равномерную сетку $[0, l] \times [0, y_l]$ на прямоугольнике с шагом Δ;
- каждая точка (x_i, y_j) проецируется на высоту $z_{ij} = \varphi(x_i, y_j)$;
- маршрут аппроксимируем *ломаной*, проходящей лишь по узлам сетки, без самопересечений, из стартового узла $(0, 0)$ в (l, y_l).

Пусть такая ломаная имеет узлы $\{(x_0, y_0), \ldots, (x_m, y_m)\}$. На каждом элементарном отрезке $\big((x_i, y_i), (x_{i+1}, y_{i+1})\big)$ длина в *трёхмерном* пространстве вычисляется по формуле

$$\Delta s_i \;=\; \sqrt{\Delta x_i^2 + \Delta y_i^2 + \Delta z_i^2}\,,$$

где $\Delta z_i = z_{i+1} - z_i$. Для приближённой оценки $\int_0^x \sqrt{1 + y'^2 + z'^2}\,d\xi$ вводим кумулятивную длину

$$s_k \;=\; \sum_{i=0}^{k-1} \Delta s_i.$$

Таким образом, дискретный аналог (1) будем интегрировать по *сегментам*, используя, например, метод средних прямоугольников или трапеций. Тогда полученное приближение можно записать

$$J_{\text{discr}}(R) \;\approx\; \sum_{k=0}^{m-1} \Big[\, \alpha_k \left(\tfrac{s_k + s_{k+1}}{2}\right) \;+\; \beta_k \Big] \,\frac{\Delta s_k + \Delta s_{k+1}}{2}, \qquad (2)$$

где $\alpha_k = \alpha(x_k, y_k)$, $\beta_k = \beta(x_k, y_k)$. Такая схема даёт более точную оценку, чем простое умножение $\alpha_k\, s_k + \beta_k$ на Δs_k.

3. Применение генетического алгоритма. Генетический алгоритм (ГА) в данной постановке основан на идеях [1], где аналогичный метод применялся к двумерной задаче, а также на общих работах по эволюционным методам [5].

Структура алгоритма следующая:

1. *Генерация начальной популяции.* Формируем N случайных ломаных на сетке, контролируя отсутствие самопересечений. Для каждого маршрута R вычисляем $J_{\text{discr}}(R)$ по формуле (2).

2. *Скрещивание (crossover).* Пары лучших маршрутов ищут общие узлы пересечения, разрезаются в них и «обмениваются» частями траекторий, формируя потомков. Потомки с более высокой стоимостью отбрасываются.

3. *Мутация.* Некоторые маршруты случайным образом «смещают» одну из узловых точек в соседний узел сетки. Если это уменьшает J_{discr} и не создаёт самопересечений, мутировавший маршрут остаётся.

4. *Селекция.* Формируем новое поколение, оставляя лучшие варианты и (реже) добавляя небольшую «долю случайности» для поддержания разнообразия.

5. *Критерий остановки.* Выполняем несколько десятков поколений или завершаем, когда улучшение лучших маршрутов прекращается на заданном числе итераций подряд.

Дополнительные детали реализации:

1. *Ускорение оценки* Δs. Поскольку при мутациях и скрещиваниях меняются не все отрезки, а лишь локальные фрагменты, возможно переиспользование ранее вычисленных Δs_i и s_k. Это существенно ускоряет ГА при большом числе поколений.

2. *Параллельная обработка.* Генерация и оценка популяции легко распараллеливаются (каждый маршрут оценивается независимо). Это даёт выигрыш в несколько раз на многопроцессорных системах.

3. *Сглаживание решения.* Полученная ломаная при желании может быть сглажена кубическим сплайном. Хотя это не уменьшает дискретный функционал, зато даёт более «гладкую» аппроксимацию реального маршрута.

4. Численные эксперименты и визуализация. Для проверки реализовали ГА на языке Python (`NumPy`, `Shapely`, `joblib` и др.). В качестве теста взяли функцию $\varphi(x, y) = \sin(5x)\sin(y)$ на области $\{0 \leqslant x \leqslant 1,\ 0 \leqslant y \leqslant 1\}$. Величины $\alpha(x, y) = 0{,}1$ и $\beta(x, y) = 0{,}5$. Шаг сетки Δ выбирался в диапазоне от 0,05 до 0,1. Число маршрутов в популяции составляло от 4 000 до 16 000, а поколений – до 40. Полученные решения сравнивались с полуаналитическим «эталоном», вычисленным с помощью вариационного метода Ритца [2, 6].

4.1. Пример с $\Delta = 0{,}1$ и популяцией 16000 особей. В одном из экспериментов брали сравнительно «крупный» шаг сетки, $\Delta = 0{,}1$, и одновременно увеличивали начальную популяцию до 16 000 особей.

Итоговый маршрут получен после 14 поколений, при этом разница в стоимости J с эталонным решением не превысила 4,3%. На рис. 1 показана визуализация полученной ломаной на трёхмерной поверхности $z = \sin(5x)\sin(y)$.

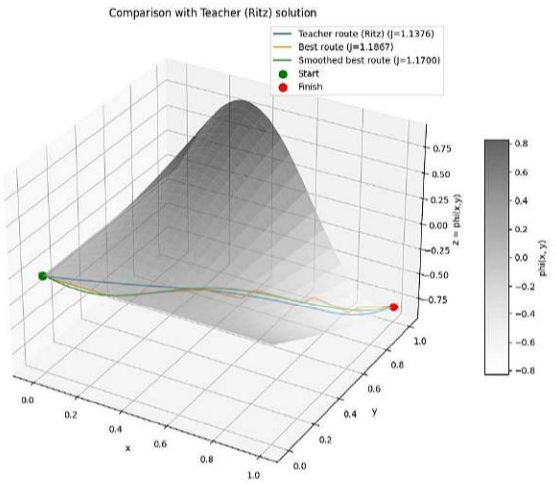


Рис. 1. Итоговая траектория при шаге сетки $\Delta = 0{,}1$ и популяции $N = 16000$

5. Заключение. Таким образом, дискретизация вариационной постановки (1) в трёхмерном пространстве и применение генетического алгоритма позволяют эффективно искать траектории с минимальной стоимостью, учитывающей протяжённость пути и рельеф. Ключевые достоинства:

- не требуется задавать гладкие условия на рельеф и выпуклость функционала;
- обеспечивается гибкость и параллелизм вычислений при большом размере популяции;
- легко расширять алгоритм на случаи с дополнительными ограничениями (запретные зоны) или с несколькими критериями (уклон, безопасность и т. п.).

В дальнейшем планируется увеличить точность (2), используя более тонкие схемы интегрирования, а также протестировать гибридные алгоритмы, совмещающие преимущества ГА и градиентных методов поиска локальных улучшений.

Литература

1. Lavrukhin M. U., Abbasov M. E. Genetic algorithm in the problem of finding the optimal trajectories // Proceedings of MaCoSEP. 2025. Accepted for publication.

2. Abbasov M. E., Gorbunova A. A. Dynamic programming algorithms for finding cost-optimal trajectory on the terrain // arXiv preprint arXiv:2503.10922. 2025. 15 p.

3. Holland J. H. Adaptation in Natural and Artificial Systems. Ann Arbor, MI: Univ. of Michigan Press, 1975. 183 p.

4. Mansfield R. P. Applying genetic algorithms to multi-Objective route planning for autonomous vehicles // journal of applied AI. 2019. Vol. 23. No 2. P. 123–137.

5. Sohrabi M., Fathollahi-Fard A. M., Gromov V. A. Genetic engineering algorithm (GEA): an efficient metaheuristic algorithm for solving combinatorial optimization problems // arXiv preprint arXiv:2309.16413. 2023.

6. Аббасов М. Э., Шарлай А. С. Поиск оптимальной по стоимости строительства траектории дороги на рельефе местности // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2021. Т. 17. № 1. С. 4–12.

УДК 519.233.33

Миронов А. В.

Имплементация и анализ мощности критериев согласия с распределением Вейбулла

Рекомендовано к публикации доцентом Гориховским В. И.

1. Введение. Исследование и анализ статистических распределений играют ключевую роль в различных областях науки и техники, таких как инженерия, экономика, медицина и экология [1–3]. Одним из наиболее широко используемых распределений для моделирования времени до отказа и анализа надежности является распределение Вейбулла. Его гибкость, обусловленная наличием параметров формы и масштаба, позволяет описывать разнообразные типы данных, что делает его универсальным инструментом в статистическом моделировании. Однако корректное применение распределения Вейбулла требует надежных методов проверки согласия эмпирических данных с теоретической моделью.

В данной статье рассматриваются различные критерии согласия, такие как критерий Колмогорова-Смирнова, критерий Андерсона-Дарлинга и другие, которые используются для проверки гипотезы о соответствии выборки распределению Вейбулла. Основное внимание уделяется сравнению их мощностей, то есть способности корректно отвергать нулевую гипотезу при наличии отклонений от предполагаемого распределения. Мощность критериев оценивается в зависимости от объема выборки, параметров распределения Вейбулла и степени отклонения от модели.

Ранее исследование критериев согласия с распределением Вейбулла проводилось в работе [4]. В данной работе было произведено исследование критериев согласия с распределением Вейбулла, однако исследовались выборки небольшого размера, а также не предоставлен исходный код критериев.

В данной работе будет представлена реализация критериев согласия с распределением Вейбулла, инструмент для проведения сравнительного анализа критериев согласия, а также будет производиться анализ критериев.

Миронов Алексей Владиславович – аспирант, Санкт-Петербургский государственный университет; e-mail: alexmir2015@yandex.ru, тел.: +7(911)146-14-40

Работа выполнена при поддержке СПБГУ (Pure ID 116636233)

Актуальность данного исследования обусловлена необходимостью выбора наиболее эффективного критерия согласия для анализа данных, что особенно важно при работе в условиях ограниченных ресурсов. Результаты проведенного анализа могут быть полезны для исследователей и практиков, занимающихся статистическим моделированием и анализом надежности, а также для разработки рекомендаций по применению критериев согласия в различных сценариях.

2. Интеграция в PySATL. Была разработанная методика и программное окружение, которые позволяют эффективно анализировать мощность критериев согласия. Это может использоваться для выбора подходящего критерия в зависимости от объема данных и ожидаемых отклонений от нулевой гипотезы. При разработке программного окружения использовались современных инструменты, таких как Python, что позволяет сделает процесс анализа доступным и гибким.

В рамках исследования были реализованы следующие критерии согласия с распределением Вейбулла, которые доступны в репозитории фреймворка $PySATL$[1] :

Kolmogorov–Smirnov (KS) [5], Cramer–von Mises (CVM) [7], Shapiro and Wilk (SW) [8], Epps-Pulley (EP) [9], Min-Toshiyuki (MT) [10], Kurtosis (ST1) [22], Skew (ST2) [22], Anderson–Darling (AD) [6], Tiku and Singh (TS) [11], Lockhart-O'Reilly-Stephens (LOS) [12], Mann-Scheuer-Fertig (MSF) [13], Ozturk and Korukoglu (OK) [14], Shapiro and Brain (SB) [15], Smith and Bain (RSB) [16], Laplace transform (LT, LT2, LT3) [17], Kullback-Leibler (KL) [18], Liao-Shimokawa (LS) [19], Watson (W) [20], Mahdi-Doostparast (MD) [21], Evans-Johnson-Green (REJG) [22]

3. Эмпирическое исследование критериев. В данной работе используются 22 критерия согласия для проверки распределения Вейбулла с коэффициентом масштаба 1 и параметром формы 5, при уровне значимости 0.95. Фреймворк PySATL позволяет проводить сравнительный анализ мощностей критериев для других гипотез согласия. Среди сравниваемых критериев некоторые являются универсальными и могут применяться для проверки не только для Вейбулла, но и других распределений, таких как экспоненциальное, Гауссово и другие. Другие критерии разработаны специально для проверки

[1] `https://github.com/PySATL/pysatl-experiment`

гипотезы согласия с распределением Вейбулла.

3.1. Организация исследования. Далее будет обсуждаться исследование. Исследование состоит из 4 этапов:

1. **Формирование альтернативных гипотез.** В качестве альтернативных гипотез использовались 7 различных распределений с разными параметрыми, которые будут представлены далее.
2. **Генерация данных.** Генерируются по 1000 выборок альтернатив размерами 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000.
3. **Расчет мощности.** Расчет мощности производился для каждой альтернативы и уровни значимости 0.95.
4. **Анализ результатов.** Полученные данные были проанализированы и сформулированы рекомендации по использованию критериев.

При проведении эксперимента использовались следующие альтернативы:

- три случая Экспоненциального распределения – $Exp(l)$: $Exp(0.5)$, $Exp(1)$, и $Exp(1.5)$, где l это обратный коэффициент масштаба;
- шесть случаем Гамма распределения – $Gamma(a, b)$: $Gamma(1; 2)$, $Gamma(2; 2)$, $Gamma(3; 2)$, $Gamma(5; 1)$, $Gamma(9; 0.5)$, и $Gamma(0.5; 1)$, где a и b параметры формы и масштаба;
- четыре случая обратного Гауссово распределения – $InvGauss(m, l)$: $InvGauss(1; 0.2)$, $InvGauss(1; 1)$, $InvGauss(1; 3)$, $InvGauss(3; 0.2)$ и $InvGauss(3; 1)$ где m и l параметры сдвига и формы;
- четыре случая Логнормального распределения – $Lognormal(t, s)$: $Lognormal(0, 0.25)$, $Lognormal(0, 1)$, $Lognormal(0, 5)$, $Lognormal(5, 0.25)$, $Lognormal(5, 1)$, $Lognormal(5, 5)$, где t и s параметры сдвига и масштаба;
- три случая распределения Райса – $Rice(n, s)$: $Rice(0, 1)$, $Rice(0.5, 1)$, $Rice(4, 1)$ где n расстояние между точкой отсчета и центром двумерного распределения и s параметр масштаба;
- четыре случая распределения Гомпертца – $Gompertz(e, b)$: $Gompertz(0.1, 1)$, $Gompertz(2, 1)$, $Gompertz(3, 1)$, $Gompertz(1, 2)$, $Gompertz(1, 3)$ где e и b параметры формы и масштаба;

- один случай распределения Вейбулла – $Weibull(l, k)$ распределения $Weibull(2, 5)$ где l и k параметры масштаба и формы;

3.2. Результаты и анализ. Для каждого критерия против каждой альтернативы была рассчитана статистическая мощность каждого. Из полученной таблицы мощностей были выбраны три самых мощных критерия для каждого размера выборки и подсчитано количество вхождений каждого критерия в список. Вычисленное количество вхождений было разделено на три группы 10–90, 100–400, 500–1000. Результат представлен в таблице 1.

Легко увидеть, что критерий KS можно выделить как самый мощный критерий среди других для гипотезы согласия с распределением Вейбулла с коэффициентом масштаба 1 и параметром формы 5. Кроме того, критерии LILLIE и LT2 демонстрируют значительную мощность, особенно при применении к размерам выборки от 10 до 100. Для размеров выборки от 100 до 500 критерий KS и LILLIE продолжают доминировать как наиболее мощные критерии. Аналогично, критерии KS, LILLIE и AD являются наиболее мощными для размеров выборки от 500 до 1000.

Таблица 1. Количество трех самых мощных критериев

Критерий	$\mathbf{n < 100}$	$\mathbf{100 \leqslant n < 500}$	$\mathbf{500 \leqslant n \leqslant 1000}$
KS	233	106	121
LILLIE	111	73	90
LT2	113	31	31
LS	20	8	15
MD	80	22	16
CVM	74	41	52
KL	52	27	36
LOS	4	6	4
AD	1	28	68
MSF	1	1	1
SB	9	1	1

4. Заключение. В данной работе был проведен анализ мощности критериев согласия для распределения Вейбулла, реализованных во фреймворк PySATL. Разработанная методика позволяет систематически оценить эффективность различных критериев согласия. Использование фреймворка PySATL обеспечило удобную и воспроизводимую среду для проведения экспериментов, включая генерацию данных, расчет статистик и визуализацию результатов.

Литература

1. Dai W., Toshimitsu H. Statistics in Medicine. Изд. 4-е. Oxford, UK: Academic Press, 2022. 795 с.
2. Lawrence R. K. The statistical approach to economics // Journal of Econometrics. 1988. Vol. 37. No 1. P. 7–26.
3. Zhan W., Rainer F., Fang A. Application Of Statistics In Engineering Technology Programs // American Journal of Engineering Education. 2010. Vol. 1. No 1. P. 65–78.
4. Krit M., Gaudoin O., Remy E. Goodness-of-fit tests for the Weibull and extreme value distributions: A review and comparative study // Communications in Statistics - Simulation and Computation. 2019. Vol. 50. No 7. P. 1888–1911.
5. Kolmogorov A. N. Sulla determinazione empirica di una legge di distribuzione // Giornale dell'Istituto Italiano degli Attuari. 1933. Vol. 4. P. 83–91.
6. Anderson T. W.; Darling D. A. A Test of Goodness-of-Fit // Journal of the American Statistical Association. 1954. Vol. 49. P. 765–769.
7. Bush M. G. A Modified Cramer-von Mises and Anderson-Darling Test for the Weibull Distribution with Unknown Location and Scale Parameters // Communications in Statistics-theory and Methods. 1983. Vol. 12. P. 2465–2476.
8. Shapiro S. S, Wilk M. B. An analysis of variance test for the exponential distribution (complete samples) // Technometrics. 1972. Vol. 14. P. 355–370.
9. Epps T. W., Pulley L. B. A Test for Normality Based on the Empirical Characteristic Function // Biometrika. 1983. Vol. 70. No 3. P. 723–726.
10. Liao M., Toshiyuki S. A New Goodness-of-Fit Test for Type-I Extreme-Value and 2-Parameter Weibull Distributions with Estimated Parameters // Journal of Statistical Computation and Simulation. 1999. Vol. 64. No 1. P. 23–48.
11. Tiku M. L., Singh M. Testing the two-parameter Weibull distribution, Communications in Statistics // Communications in Statistics – Theory and Methods. 1981. Vol. 10. No 9. P. 907–918.

12. Lockhart R. A., O'Reilly F., Stephens M. A. Tests for the extreme-value and Weibull distributions based on normalized spacings // Naval Research Logistics Quarterly. 1986. Vol. 33. P. 413–421.

13. Mann N. R., Scheuer E M., Fertig K. W. A new goodness-of-fit test for the two-parameter Weibull or extreme-value distribution // Communications in Statistics. 1973. Vol. 2. P. 383–400.

14. Ozturk A. On the W Test for the Extreme Value Distribution // Biometrika. 1986. Vol. 73. No 3. P. 738–740.

15. Brain C. W., Shapiro S. S. A regression test for exponentiality: Censored and complete samples // Technometrics. 1983. Vol. 25. P. 69–76.

16. Smith R. M., Bain L. J. Correlation type goodness-of-fit statistics with censored sampling // Communications in statistics. 1976. Vol. 5. P. 119–132.

17. Ebner B. Fischer A., Henze N., et al. Goodness-of-fit tests for the Weibull distribution based on the Laplace transform and Stein's method // Ann Inst Stat Math. 2023. Vol. 75. P. 1011–1038.

18. Song K. S. Goodness-of-fit tests based on Kullback-Leibler discrimination information // IEEE Transactions on Information Theory. 2002. Vol. 48. No 5. P. 1103–1117.

19. Liao M., Shimokawa T. A new goodness-of-fit test for type-I extreme value and 2-parameter Weibull distributions with estimated parameters // Journal of Statistical Computation and Simulation. 1999. Vol. 64. P. 23–48.

20. Henze N., Meintanis S. G., Ebner B. Goodness-of-fit tests for the Gamma distribution based on the empirical Laplace transform // Communications in Statistics – Theory and Methods. 2012. Vol. 41. No 9. P. 1543–1556.

21. Doostparast M. Goodness-of-?t tests for weibull populations onthe basis of records // Stat Theory Appl. 2015. Vol. 14. No 3. P. 289–300.

22. Evans J. W., Johnson R. A., Green D. W. Two and three parameter Weibull goodness-of-fit tests. Madison: Forest Products Laboratory. 1989. P. 34.

УДК 004.85+519.237.8

Мошкин Л. С.

Использование аппроксимационно-оценочных критериев для модификации алгоритма HDBSCAN

Рекомендовано к публикации старшим преподавателем Ореховым А. В.

1. Введение. Алгоритмы кластеризации, основанные на оценке плотности распределения данных, формируют кластеры в n-мерном евклидовом пространстве $\mathbb{E}^n$ как множества повышенной концентрации точек, отделяя их от разреженных областей. Эти алгоритмы обнаруживают закономерности распределения данных исключительно на основе их пространственного местоположения.

Алгоритм кластеризации DBSCAN (англ. Density-based spatial clustering of applications with noise) основан на плотности распределения данных [1]. Для его выполнения необходимо задать два глобальных параметра: ε – радиус окрестности и minPts – минимальное число точек, которые должны принадлежать ε-окрестности. В балансе параметров minPts и ε заключается сложность применения DBSCAN: реальные наборы данных содержат кластеры разной плотности с границами разной степени размытости. Главные недостатки DBSCAN – неспособность соединять кластеры через проемы, и, наоборот, способность связывать явно различные кластеры через плотно «населенные» перемычки. Отчасти поэтому при увеличении n проявляется «проклятие размерности»: чем больше n, тем больше мест, где могут случайно возникнуть проемы или мосты. Для преодоления этих недостатков был предложен алгоритм HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise), который строит кластеры так же, как и DBSCAN, но использует «изменяющиеся» расстояния, позволяя формировать кластеры с различной плотностью данных [2].

Чтобы выделить кластеры, необходимо найти «сгустки данных» с более высокой плотностью распределения среди всех остальных дан-

Мошкин Лев Станиславович – студент, Санкт-Петербургский государственный университет; e-mail: lev-moshkin@yandex.ru, тел.: +7(951)352-11-08

ных, рассматриваемых как шум. Основой алгоритма HDBSCAN является иерархическая агломеративная кластеризация методом одиночной связи (single linkage). Этот агломеративный метод кластеризации чувствителен к шуму; одна точка данных «в неправильном месте» может действовать как «мост» между «сгустками данных», «склеивая» их в один кластер. Для увеличения робастности HDBSCAN необходимо найти способ «максимально понизить» уровень плотности шума. Кроме того, одной из основных проблем кластерного анализа является определение предпочтительного числа кластеров. В общем случае эта задача не решена [3, 4], а для иерархических агломеративных методов кластеризации ее решение непосредственно связано с моментом остановки процесса [5, 6]. В данной работе предлагается использовать аппроксимационно-оценочные критерии для модификации HDBSCAN и определения оптимального числа кластеров.

2. Алгоритм HDBSCAN. В алгоритме для формирования кластеров используется идея плотности пространства объектов, которая реализуется с помощью расстояния взаимной достижимости.

Обозначим через $d(x_p, x_q)$ расстояние между двумя объектами $x_p, x_q \in X$ метрического пространства $\{X, d\}$.

Определение 1. Базовым расстоянием $\mathrm{core}_k(x_i)$ для объекта $x_i \in X$ называется расстояние от x_i до его k ближайших соседей.

Определение 2. Расстояние взаимной достижимости между двумя объектами $x_p, x_q \in X$ определяется как $d_{\mathrm{mreach}}(x_p, x_q) = \max\{\mathrm{core}_k(x_p), \mathrm{core}_k(x_q), d(x_p, x_q)\}$.

Определение 3. Устойчивость кластера C_i определяется как $\sigma(C_i) = \sum_{x_j \in C_i}(\lambda_{\max}(x_j, C_i) - \lambda_{\min}(x_j, C_i))$, где $\lambda_{\max}(x_j, C_i)$ – максимальное расстояние взаимной достижимости, при котором объект x_j все еще принадлежит кластеру C_i, $\lambda_{\min}(x_j, C_i)$ – минимальное расстояние взаимной достижимости, при котором объект x_j начинает принадлежать кластеру C_i.

Алгоритм HDBSCAN включает следующие основные этапы:

1. Преобразование пространства данных.
2. Построение минимального остовного дерева.
3. Построение кластерной иерархии связанных компонент.
4. Сжатие иерархии кластеров.
5. Извлечение стабильных кластеров из сжатого дерева графа.

На первом этапе метрическое расстояние между объектами заменяется на расстояние взаимной достижимости, которое способствует отделению объектов, содержащих небольшое число соседей, от объектов, находящихся в областях высокой плотности.

Следующий этап включает построение неориентированного простого взвешенного графа, вершинами которого являются объекты пространства, а ребра соединяют каждую пару объектов и имеют вес, равный расстоянию взаимной достижимости между этими объектами. Для этого графа ищется минимальное остовное дерево.

На третьем этапе ребра минимального остовного дерева сортируются в порядке неубывания. Тем самым формируется иерархия кластеров по методу одиночной связи.

На четвертом этапе осуществляется проход по полученной кластерной иерархии, начиная с кластера, содержащего все элементы. Если в результате разбиения один из дочерних кластеров имеет размер меньше минимального, то объекты, входящие в него, помечаются как шумовые при соответствующем пороге расстояния взаимной достижимости. Другой дочерний кластер сохраняет свой вид. Иначе разбиение считается истинным и остается в иерархии.

Извлечение стабильных кластеров из сжатого дерева графа содержит проход по иерархии от малых кластеров к большим. Если сумма устойчивости дочерних кластеров больше устойчивости родительского кластера, мера устойчивости родителя становится равной сумме мер прямых потомков. В ином случае выбирается родительский кластер вместо всех его потомков.

3. Аппроксимационно-оценочные критерии. Одной из основных проблем при использовании методов агломерационной кластеризации является вычисление предпочтительного количества кластеров, решение этой задачи непосредственно связано с определением момента остановки процесса [5, 6].

Пусть $F_1, \ldots, F_{m-1}$ – минимальные расстояния между кластерами, полученными на 1-й, 2-й и т. д. итерациях агломеративной кластеризации. Данная последовательность является монотонной для всех агломеративных алгоритмов, кроме центроидного [7, 8]. Набор $y_1, \ldots, y_{m-1}$ получен с использованием преобразования $y_i = F_i + qi$, где q – коэффициент тренда, i – номер итерации алгоритма кластеризации. Также положим $y_j = 0$, это можно получить

при помощи преобразования

$$y_j = y_j - y_j, y_{j+1} = y_{j+1} - y_j, \ldots, y_{j+k} = y_{j+k} - y_j.$$

Квадратичная ошибка аппроксимации функцией f натуральных узлов $y_j, \ldots, y_{j+k}$ имеет вид

$$\delta_f^2(k) = \sum_{i=j}^{j+k} \left(f(i) - y_i\right)^2.$$

В случае линейной аппроксимации используется функция $f(i) = ai + b$, при экспоненциальной аппроксимации – $f(i) = ge^i + h$ и неполной параболической аппроксимации – $f(i) = ci^2 + d$. Коэффициенты a, b, c, d, g, h вычисляются методом наименьших квадратов.

Обозначим через $\delta^2(k)$ квадратичную форму разности линейной и нелинейной аппроксимаций, которая меняет свой знак в момент, когда характер увеличения числовых значений последовательности y_i изменяется с линейного типа на нелинейный.

В общем случае аппроксимационно-оценочный критерий формулируется следующим образом: процесс агломеративной кластеризации считается завершенным на k-й итерации, если для узлов $y_{j-1}, y_j, \ldots, y_{j+k-1}$ справедливо неравенство $\delta^2(k) \leqslant 0$, а для набора точек $y_j, y_{j+1}, \ldots, y_{j+k}$ выполняется $\delta^2(k) > 0$ [5, 6].

Процесс кластеризации завершается с помощью описанного выше критерия, который оценивает скачки монотонно возрастающей последовательности $y_1, \ldots, y_{m-1}$. Величина скачка, достаточного для остановки процесса, зависит от чувствительности критерия остановки, она задается с помощью неотрицательного коэффициента q. Чем выше значение q, тем ниже чувствительность критерия остановки процесса кластеризации, он имеет наибольшую чувствительность при $q = 0$, в этом случае при кластеризации будет получено наибольшее количество кластеров. При увеличении q чувствительность критерия может быть уменьшена так, что процесс будет продолжаться до тех пор, пока все объекты не будут объединены в один кластер. Отрезок устойчивой кластеризации $Q_i = [\alpha_i, \beta_i]$ содержит диапазон значений q, при которых будет получен один и тот же результат кластеризации.

В общем случае, последовательность отрезков устойчивой кластеризации для различных значений коэффициента q обозначается

как $Q_1, \ldots, Q_{e-2}, Q_{e-1}, Q_e$, где Q_e – набор значений коэффициента q, в котором все объекты объединены в один кластер.

В работе [9] сформулирована гипотеза о том, что предпочтительное число кластеров формируется при $q \in Q_{e-2}$.

Основным мотивом для формулировки этой гипотезы было то, что цепной эффект, при котором уже сформированные кластеры добавляются один за другим к какому-то другому кластеру, проявляется на отрезке устойчивой кластеризации Q_{e-1}.

4. Модификация алгоритма HDBSCAN. В вычислительных экспериментах вместо кластерной иерархии на основе минимального остовного дерева и метода single linkage строилась таксономия модельных данных с помощью различных методов иерархической агломеративной кластеризации. Предпочтительное число кластеров определялось исходя из $(e-2)$-гипотезы. Момент остановки процесса кластеризации вычислялся при помощи параболических и экспоненциальных аппроксимационно-оценочных критериев, квадратичные формы которых приведены в работе [10].

В качестве модельного набора данных использовались «Ирисы Фишера» [11]. Этот набор данных включает три класса ирисов, каждый класс описывается четырьмя характеристиками: длина/ширина чашелистика и длина/ширина лепестка. Оценка качества кластеризации производилась при помощи V-меры [12].

Показано, что при кластеризации «ирисов Фишера» модифицированным методом (MHDBSCAN), V-мера не отличается при использовании экспоненциального и неполного параболического критериев по пяти узлам аппроксимации для всех случаев агломеративной кластеризации, кроме метода Уорда (см. таблицу). Лучший результат получен для этого метода вместе с параболическим аппроксимационно-оценочным критерием.

Таблица. V-мера при использовании различных методов кластеризации

Агломеративный метод	**Параболический критерий**	**Экспоненциальный критерий**
Одиночная связь	0,08327	0,08327
Полная связь	0,66746	0,66746
Средняя связь	0,70457	0,70457
Центроидный метод	0,70457	0,70457
Метод Уорда	0,77766	0,63434

Результаты вычислительных экспериментов приведены на рис. 1.

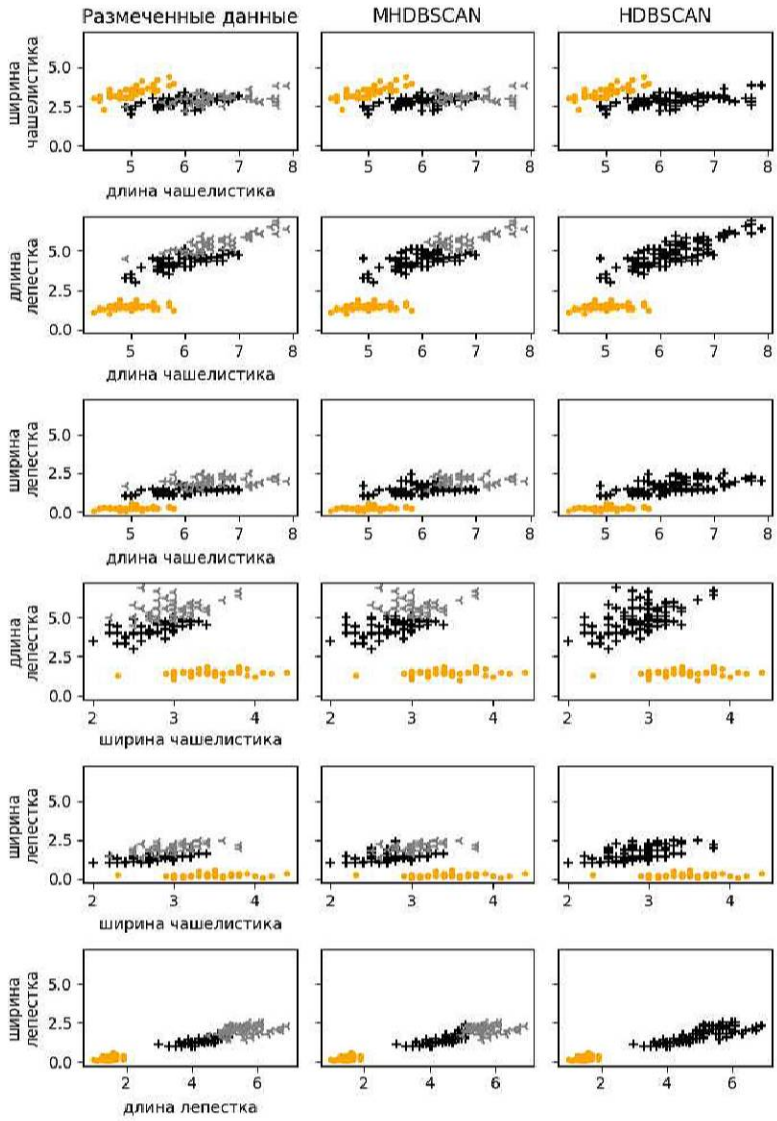


Рис. 1. Кластеризация «ирисов Фишера» модифицированным и немодифицированным алгоритмами HDBSCAN

V-мера результатов кластеризации «ирисов Фишера» при помощи MHDBSCAN с иерархическим методом Уорда и неполным параболическим критерием равна 0,77766, а V-мера стандартного HDBSCAN для того же набора отличается всего на несколько сотых (0,73368), но модифицированный алгоритм выделяет три кластера, а не два, что соответствует реальной структуре данных.

Для проведения вычислительных экспериментов был написан код на языке Python 3.12 в среде разработки Visual Studio Code. Набор данных «Ирисы Фишера» взят из библиотеки scikit-learn (sklearn.datasets.load_iris). Также из этой библиотеки была взята реализация метода HDBSCAN (sklearn.cluster.HDBSCAN), которая имеет два основных гиперпараметра – минимальный размер кластера (min_cluster_size) и число соседей для расчета расстояния взаимной достижимости (min_samples).

В методе MHDBSCAN, который принимал на вход те же гиперпараметры, что и HDBSCAN, для реализации расчета расстояния между объектами использовалась функция scipy.spatial.distance.pdist с евклидовой метрикой. Построение дендрограммы осуществлялось с использованием функции scipy.cluster.hierarchy.linkage, которой в качестве аргумента передавалась построенная матрица расстояний взаимной достижимости. Определение момента остановки осуществлялось с помощью процедуры-функции, имеющей входными параметрами последовательность минимальных расстояний между кластерами и тип аппроксимационно-оценочного критерия с указанием количества используемых узлов. Возвращаемым значением является номер шага, на котором следует остановить кластеризацию в соответствии с $(e-2)$-гипотезой. Получение меток кластеров производилось вызовом функции scipy.cluster.hierarchy.fcluster. Кластеры, которые содержали меньше элементов, чем заданный минимальный размер, определялись как шум.

Литература

1. Ester M., Kriegel H. P., Sander J., Xu X. A density-based algorithm for discovering clusters in large spatial databases with noise // Proceedings of the Second International Conference on Knowledge Discovery and Data Mining. 1996. P. 226–231.

2. Campello R. J. G. B., Moulavi D., Sander J. Density-based clustering based on hierarchical density estimates // Lecture Notes in Computer Science. 2013. P. 160–172.

3. Everitt B. S. Cluster Analysis. Chichester, West Sussex, UK: John Wiley & Sons Ltd. Press, 2011. 330 p.

4. Duda R. O., Hart P. E., Stor D. G. Pattern Classification. New York, Chichester: Wiley Press, 2001. 654 p.

5. Орехов А. В. Марковский момент остановки агломеративного процесса кластеризации в евклидовом пространстве // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2019. Т. 15. № 1. С. 76–92.

6. Орехов А. В., Васильев Е. В. Бинарные метрические деревья и иерархия вложенных кластеров // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2024. Т. 20. № 4. С. 487–499.

7. Lance G. N., Williams W. T. A general theory of classificatory sorting strategies. Hierarchical systems // The Computer Journal. 1967. Vol. 9. No 4. P. 373–380.

8. Milligan G. W. Ultrametric hierarchical clustering algorithms // Psychometrika. 1979. Vol. 44. P. 343–346.

9. Orekhov A. V. Agglomerative method for texts clustering // Proceedings of the 5th International Conference on Internet Science (INSCI 2018) / Ed. by S. Bodrunova. Switzerland. Springer: Cham, 2019. Vol. 11551. P. 19–32.

10. Orekhov A. V. Quasi-deterministic processes with monotonic trajectories and unsupervised machine learning // Mathematics. 2021. Vol. 9. No 18. P. 1–26.

11. Fisher R. A. The use of multiple measurements in taxonomic problems // Annals of Eugenics. 1936. Vol. 7. P. 179–188.

12. Rosenberg A., Hirschberg J. V-Measure: A conditional entropy-based external cluster evaluation measure // Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. 2007. P. 410–420.

УДК 004.89

Мурадян Д. С., Алимов П. Г.

Реализация и внедрение архитектуры Gated Recurrent Unit в пакет нейросетевой аппроксимации дифференциальных уравнений DEGANN

Рекомендовано к публикации доцентом Гориховским В. И.

1. Введение. Дифференциальные уравнения (ДУ) играют ключевую роль в самых разных областях науки и техники. Решение таких уравнений – задача, требующая больших вычислений. С усложнением и увеличением размерности задачи эффективность традиционных численных методов решения дифференциальных уравнений снижается, что приводит к увеличению времени вычислений и потере точности [1].

В связи с указанными проблемами, возникает необходимость в поиске альтернативных методов решения таких задач. Одним из них является использование нейронных сетей, которые могут аппроксимировать функции, описывающие ДУ. Такой подход используется в пакете нейросетевой аппроксимации ДУ DEGANN [2].

На данный момент функциональность пакета DEGANN ограничена одной архитектурой НС – многослойным перцептроном (MLP – Multi-Layer Perceptron). Эта архитектура может быть недостаточно эффективна в некоторых задачах аппроксимации ДУ, так как MLP не достаточно хорошо улавливает последовательные зависимости в данных [3]. В отличие от MLP, рекуррентные нейронные сети (RNN – Recurrent Neural Network) [4] специально предназначены для работы с последовательностями, которые очень важны в анализе нелинейных ДУ.

Мурадян Денис Степанович – студент, Санкт-Петербургский государственный университет, e-mail: muradyan.denis@inbox.ru, тел.: +7(905)204-57-23;

Алимов Павел Геннадьевич – магистрант, Санкт-Петербургский государственный университет, e-mail: st076209@student.spbu.ru, тел.: +7(904)518-13-99;

Работа выполнена при финансовой поддержке СПБГУ, грант № 116636233

Однако RNN имеют известную проблему затухания и взрыва градиента, из-за чего обучение становится затруднительным на длинных последовательностях [5]. Архитектура (GRU – Gated Recurrent Unit) является усовершенствованной версией RNN и частично решает эту проблему благодаря использованию элементов управления потоком данных.

Внедрение GRU в пакет DEGANN может помочь оптимизировать и улучшить качество аппроксимации ДУ.

2. Реализация и интеграция в пакет. Для реализации архитектуры `Gated Recurrent Unit` в пакете DEGANN разработан класс `TensorflowGRUNet`, который наследуется от `tf.keras.Model`.

Первоначально создаётся список GRU-слоёв, с заданными функциями активации (основная – *tanh*, рекуррентная – *sigmoid*) и инициализаторами весов и смещений. Последовательная обработка входных данных этими слоями передаётся в полносвязный выходной слой с линейной активацией, что обеспечивает регрессионное предсказание.

Логика компиляции модели выполняется с помощью реализованного метода `custom_compile`, который отвечает за настройку оптимизатора, функции потерь, метрик и т. д. Метод `to_dict`, экспортирует конфигурацию модели в виде словаря для дальнейшей интеграции с другими компонентами DEGANN. В методе `call` реализован прямой проход через нейросеть, а вычисление градиентов производится стандартными средствами TensorFlow. Разработанная оболочка обеспечивает более гибкую настройку и унифицированный интерфейс для работы модели в системе.

Основным элементом реализации является класс `TensorflowGRUNet`, который осуществляет следующее этапы:

- построение и инициализация GRU-слоёв, каждый из которых имеет число нейронов, определяемое параметром **block_size**;
- настройка ключевых параметров:
  - **input_size** – размер входных данных (количество признаков);
  - **output_size** – размер выхода;
  - **activation_func** и **recurrent_activation** – функции активации для основного и рекуррентного состояний (по умолчанию *tanh* и *sigmoid*);

– **weight** и **biases** – инициализаторы весов и смещений;

- организация прямого прохода данных через сеть с помощью метода `call`. В этом методе входные данные последовательно обрабатываются всеми GRU-слоями, а затем результат передается в выходной полносвязный слой с линейной активацией, что оптимально для регрессионных задач.

Интеграция архитектуры GRU в DEGANN осуществляется посредством параметра `net_type`, передаваемого при создании модели через базовый класс `IModel`. При установке `net_type="GRUNet"` внутри `IModel` вызывается метод, создающий экземпляр класса `TensorflowGRUNet`. Такой подход позволил сохранить единообразный интерфейс работы с моделями различных типов (например, с DenseNet или другими), не нарушая общую модульную структуру проекта. Это обеспечивает возможность использования стандартных методов обучения, тестирования и экспорта моделей.

3. Валидация решения. В данном разделе представлено экспериментальное сравнение эффективности двух архитектур, реализованных в DEGANN: GRUNet (на основе рекуррентных слоёв GRU) и DenseNet (традиционный многослойный перцептрон, MLP).

3.1. Параметры эксперимента. Цель эксперимента – сравнить сходимость двух архитектур при одинаковых гиперпараметрах (3 слоя, 30 нейронов в каждом, функция потерь MSE, метрики MAPE, R^2) и фиксированных временных промежутках обучения.

Конфигурация тестирующей системы: CPU: Intel i5 1240p | GPU: Iris Xe Graphics G7 80EUs | RAM: 16 gb.

3.2. Выбор тестовых функций и их обоснование. Для проверки моделей были выбраны следующие функции, встроенные в DEGANN, которые представляют собой достаточно сложные для аппроксимации зависимости:

- функция $f_{\sin}(x) = \sin(10x)$ демонстрирует периодическую нелинейность, что позволяет проверить способность модели захватывать колебательные процессы;
- функция $f_{\text{hyperbol}}(x) = \dfrac{x^2 + 0{,}5}{x + 0{,}1}$ обладает неоднородной зависимостью, где квадратичный рост сочетается с особенностями в малых значениях x, что требует от модели адаптивного поведения в различных областях определения;

- функция $f_{\text{hardsin}}(x) = \sin\Big(\ln\big(x^{\sin(10x)}\big)\Big)$, включающая логарифмическое преобразование и экспоненциальные эффекты, ещё более усложняет задачу аппроксимации за счёт усиленной нелинейности.

Такой набор тестовых функций позволяет всесторонне оценить эффективность сравниваемых архитектур, поскольку каждая из них предъявляет уникальные требования к способности модели моделировать сложные, неоднородные зависимости.

3.3. Параметры моделей. Для каждой архитектуры эксперименты проводятся при фиксированном времени обучения. В частности, результаты тестирования собираются для трёх вариантов времени обучения: 15, 30 и 45 секунд. При каждом варианте фиксируется функция потерь – Mean Squared Error (MSE), метрики – Mean Absolute Percentage Error (MAPE) и коэффициент детерминации (R^2), а также затраты по памяти.

Таким образом, основная задача эксперимента состоит в сравнении показателей качества аппроксимации для GRUNet и DenseNet при идентичных условиях обучения. При этом важно отметить, что использование одинаковых гиперпараметров позволяет провести объективное сравнение, где разница в результатах обусловлена внутренней структурой моделей, а не различиями в настройках.

3.4. Результаты эксперимента. В таблицах 1–3 приведены результаты эксперимента для каждого из выбранных интервалов времени обучения.

Таблица 1. Результаты эксперимента (время обучения 15 секунд)

Архитектура	**Тестовая функция**	**Память (MB)**	**Loss MSE**	**MAPE**	R^2
GRUNet	sin	111,36	0,003	4,149	0,994
GRUNet	hardsin	112,45	0,045	14,647	0,738
GRUNet	hyperbol	112,46	0,002	1,536	0,993
DenseNet	sin	35,00	0,091	23,095	0,812
DenseNet	hardsin	41,77	0,017	10,100	0,902
DenseNet	hyperbol	33,99	0,051	13,346	0,819

Таблица 2. Результаты эксперимента (время обучения 30 секунд)

Архитектура	**Тестовая функция**	**Память (MB)**	**Loss MSE**	**MAPE**	R^2
GRUNet	sin	115,39	0,003	4,169	0,994
GRUNet	hardsin	112,62	0,004	4,008	0,976
GRUNet	hyperbol	116,27	0,005	3,508	0,980
DenseNet	sin	37,52	0,012	7,509	0,975
DenseNet	hardsin	44,78	0,005	5,101	0,971
DenseNet	hyperbol	36,95	0,011	5,989	0,961

Таблица 3. Результаты эксперимента (время обучения 45 секунд)

Архитектура	**Тестовая функция**	**Память (MB)**	**Loss MSE**	**MAPE**	R^2
GRUNet	sin	120,59	0,003	4,101	0,995
GRUNet	hardsin	116,84	0,002	3,158	0,987
GRUNet	hyperbol	117,79	0,002	1,174	0,992
DenseNet	sin	38,98	0,007	7,339	0,985
DenseNet	hardsin	47,71	0,006	5,741	0,962
DenseNet	hyperbol	47,98	0,004	3,138	0,987

3.5. Анализ. Средние значения метрик MAPE, коэффициента детерминации (R^2) и потребляемой памяти для каждой архитектуры и временного интервала следующие:

- *время обучения 15 секунд:*
  GRUNet: MAPE = 6,777, R^2 = 0,908, память = 112,09 MB;
  DenseNet: MAPE = 15,514, R^2 = 0,844, память = 36,92 MB;

- *время обучения 30 секунд:*
  GRUNet: MAPE = 3,895, R^2 = 0,983, память = 114,76 MB;
  DenseNet: MAPE = 6,199, R^2 = 0,969, память = 39,75 MB;

- *время обучения 45 секунд:*
  GRUNet: MAPE = 2,811, R^2 = 0,991, память = 118,41 MB;
  DenseNet: MAPE = 5,406, R^2 = 0,978, память = 44,89 MB.

Анализ проведённых экспериментов демонстрирует, что архитектура GRUNet (по метрикам MAPE и коэффициенту детерминации (R^2)) при идентичных гиперпараметрах и фиксированном времени обучения обеспечивает более высокое качество аппроксимации целевых функций по сравнению с DenseNet.

4. Заключение. Несмотря на несколько более высокие затраты памяти, разница в потреблении ресурсов (порядка 80 МБ) не является значительной для пользователей в большинстве практических

конфигураций, в отличие от времени. GRUNet демонстрирует более быструю сходимость, обеспечивая лучшее качество аппроксимации за меньшее время. Это делает данную архитектуру перспективной для задач аппроксимации дифференциальных уравнений в рамках пакета DEGANN, где важна эффективность обучения при сохранении высокой точности результатов.

Литература

1. Hairer E., Wanner G. Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems. Heidelberg: Springer-Verlag, 1996. 629 с.

2. Пакет нейросетевой аппроксимации дифференциальных уравнений // DEGANN [Электронный ресурс]. URL `https://pypi.org/project/degann/` (дата обращения: 10.03.2025).

3. Chen R.T.Q., Rubanova Y., Bettencourt J., Duvenaud D. Neural Ordinary Differential Equations // Advances in Neural Information Processing Systems. 2018. No 31. P. 6571–6583.

4. Stryker C. What is a Recurrent Neural Network? // IBM [Электронный ресурс]. 2024. URL `https://www.ibm.com/think/topics/recurrent-neural-networks` (дата обращения: 04.02.2025).

5. Mosquera C., Van Houdt G., Napoles G. A review on the long short-term memory model // Artificial Intelligence Review. 2020. No 53. P. 5929–5955.

УДК 004.93'1

Петрова А. М.

Методы распознавания различных видов продуктов питания на изображениях

Рекомендовано к публикации доцентом Блекановым И. С.

1. Введение. С развитием и совершенствованием технологий люди все чаще используют их для отслеживания состояния своего здоровья [1]. С помощью машинного обучения и методов обработки изображений теперь возможно автоматизировать процесс контроля за рационом питания. В данной работе рассматривается применение нейросетевых моделей для распознавания изображений продуктов питания в холодильнике, а также способы подготовки специального набора данных. Задача усложняется разнообразием возможных классов продуктов, а также небольшим количеством доступных специализированных датасетов.

Задача детекции в компьютерном зрении – одна из самых часто встречающихся на практике [2, 3]. Существует несколько типов методов для эффективного решения данной задачи [4]. Детекторы подразделяются на два вида: одноэтапные и двухэтапные, которые в свою очередь могут использовать заранее определенные ограничивающие рамки (bounding box) или предсказывать класс без их помощи.

В данной работе одними из выбранных современных нейросетевых архитектур являются модели YOLOv8 [5, 6] и YOLOv11 [7], основное преимущество которых – скорость детекции в реальном времени и точность детекции небольших объектов, достигаемые благодаря подходу, при котором модель решает задачу детекции без выделения областей интереса (proposal regions), сразу предсказывая координаты ограничивающей рамки и класс.

2. Предлагаемое решение. На рис. 1 представлена схема решения, состоящая из следующих компонентов:

1. *Получение фото* с помощью камеры мобильного телефона.

Петрова Алиса Михайловна – студент, Санкт-Петербургский государственный университет; e-mail: st107966@student.spbu.ru, тел.: +7(929)703-89-44

2. *Предобработка изображения.* Во время обучения моделей в качестве аугментации использовались размытие по Гауссу и замена 0,5% пикселей на шум, в тестовом режиме изображения уменьшаются до размера 640 × 640 пикселей.

3. *Детекция* продуктов питания.

4. *Классификация* найденных объектов.

5. *Рекомендация рецептов* на основе полученного списка доступных продуктов.

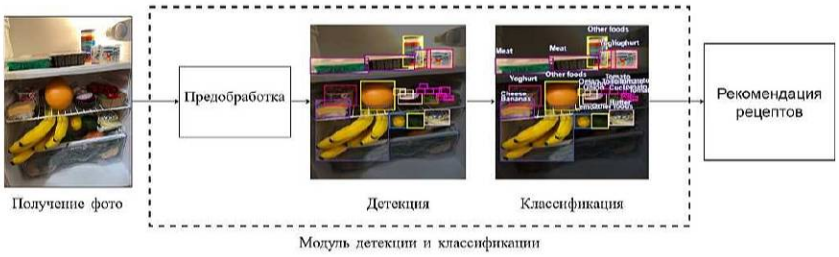


Рис. 1. Схема процесса решения задачи

В дополнение к вышеописанным моделям для решения задачи также используется модель RT-DETR [8], которая является комбинацией моделей DETR и YOLO. Этот подход превосходит по качеству модели, работающие в реальном времени, но работает дольше и хуже детектирует маленькие объекты. Во время обучения моделей были использованы гиперпараметры, представленные в таблице 1.

Таблица 1. Параметры обучения моделей

Количество эпох обучения	10
Размер батча	8
Коэффициент скорости обучения	0,01

3. Подготовка и построение набора данных. В работе были скомбинированы следующие данные:

1. Собственные данные, собранные вручную и размеченные с помощью бесплатных онлайн-инструментов [9]. Были также использованы фотографии полок в супермаркете с учетом разбиения на категории продуктов (сыры, молочные продукты и т. д.) с предположением, что обучение модели на картинках, содержащих только один класс предсказываемых объектов, должно помочь ей лучше выучить их возможное разнообразие.

2. Готовый датасет [10] с изображениями различных продуктов питания, расположенных в холодильнике.

3. Датасет [11] со снимками подносов с обедами из школьной столовой, для которого была вручную произведена нормализация аннотаций, требуемая моделями YOLO.

Таким образом, был построен оригинальный датасет с общим количеством классов 42 и количеством изображений 959, из них 805 использовались для обучения, 103 – для валидации и 51 – для теста.

4. Эксперимент. Цель эксперимента – сравнение нейросетевых моделей YOLOv8, YOLOv11, RT-DETR по следующим качественным метрикам:

1. Precision – сколько предсказаний были правильными.

2. Recall – способность модели идентифицировать все присутствующие на изображении классы.

3. mAP50 – средняя точность, посчитанная используя метрику IoU (intersection over union) с порогом 0,5.

4. mAP50-95 – средняя точность, посчитанная с использованием IoU и порогами, изменяющимися в промежутке от 0,5 до 0,95.

5. Fitness – взвешенная комбинация метрик 1–4:

$$\text{Fitness} = 0{,}1 \cdot \text{mAP50} + 0{,}9 \cdot \text{mAP50-95}.$$

Метрики моделей приведены в таблицах 2, 3. Графики на рис. 2–4 показывают снижение ошибки на обучении и валидации без значительного расхождения – модели обучаются корректно.

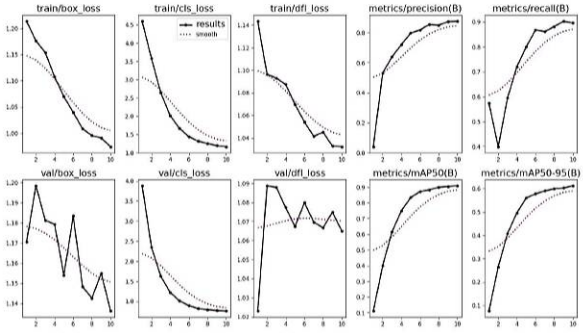


Рис. 2. Метрики модели YOLOv8

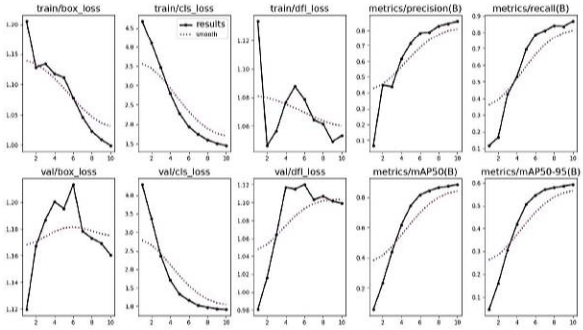


Рис. 3. Метрики модели YOLOv11

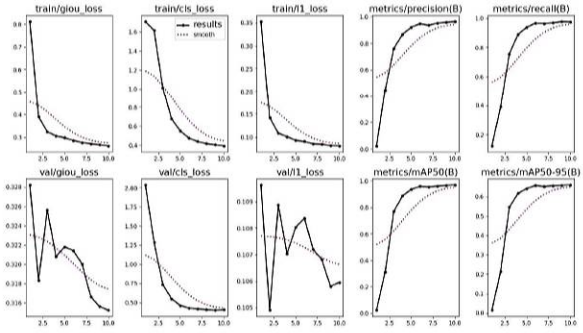


Рис. 4. Метрики модели RT-DETR

1. `train/box_loss` и `val/box_loss` – ошибки обучения и валидации (потери) по эпохам.
2. `train/cls_loss` и `val/cls_loss` – потери, связанные с классификацией.
3. `train/dfl_loss` и `val/dfl_loss` – потери, связанные с точностью предсказаний ограничивающих рамок.
4. `metrics/precision(B)` и `metrics/recall(B)` – точность и полнота модели.
5. `metrics/mAP50(B)` и `metrics/mAP50-95(B)` – среднее значение точности (Mean Average Precision)

Таблица 2. Скорости для всех моделей (ms)

Model	Preprocess	Inference	Loss	Postprocess
YOLOv8	0,8823	5,3588	0,0012	8,9678
YOLOv11	**0,2964**	**3,8001**	0,0006	5,6463
RT-DETR	0,4018	20,7346	**0,0006**	**1,4040**

Таблица 3. Метрики для всех моделей

Model	Precision	Recall	mAP50	mAP50-95	Fitness
YOLOv8	0,8855	0,8867	0,9110	0,6102	0,6403
YOLOv11	0,8281	0,8624	0,8759	0,5902	0,6188
RT-DETR	**0,9652**	**0,9739**	**0,9736**	**0,6617**	**0,6929**

5. Заключение. Построено решение (рис. 1) задачи детекции с помощью современных нейросетевых моделей. Поставленный эксперимент по распознаванию продуктов питания на изображениях из собственных данных показал, что архитектура RT-DETR примерно на 12,8% превосходит модель YOLOv11 по точности на вышеописанных качественных метриках. Однако модель YOLOv11 показала лучшую скорость работы во время предобработки входного изображения и распознавания объектов.

В дальнейшем планируется модифицировать модуль детекции и классификации (рис. 1) и добавить рекомендацию рецептов на основе найденных на фото продуктов.

Литература

1. Mereddy D. Vision-based automatic groceries tracking system – smart homes, 2023 [Электронный ресурс]: `URL:https://arxiv.org/pdf/2312.12486` (дата обращения: 17.12.24).

2. Сермягин Н. В. Детектирование людей в видеопотоке с применением детектора движения и сверточной нейронной сети // Процессы управления и устойчивость. 2021. Т. 8. № 1. С. 321–325.

3. Ворошилов А. А. Применение искусственных нейронных сетей в задаче оптического распознавания математических выражений // Процессы управления и устойчивость. 2018. Т. 5. № 1. С. 272–276.

4. Neha F., Bhati D., Shukla D. K., Amiruzzaman M. From classical techniques to convolution-based models: A review of object detection algorithms, 2024 [Электронный ресурс]: URL:https://arxiv.org/pdf/2412.05252 (дата обращения: 12.01.25).

5. Reis D., Kupec J., Hong J., Daoudi A. Real-time flying object detection with YOLOv8, 2023 [Электронный ресурс]: URL:https://arxiv.org/pdf/2305.09972 (дата обращения: 15.01.25).

6. Бендриковский А. Я. Обнаружение и классификация объектов на изображении с помощью нейросетевого алгоритма YOLO // Процессы управления и устойчивость. 2017. Т. 4. № 1. С. 326–331.

7. Khanam R., Hussain M. YOLOv11: An overview of the key architectural enhancements, 2024 [Электронный ресурс]: URL:https://arxiv.org/pdf/2410.17725 (дата обращения: 13.01.25).

8. Zhao Y., Lv W., Xu S., Wei J., Wang G., Dang Q., Liu Y., Chen J. DETRs beat YOLOs on real-time object detection, 2023 [Электронный ресурс]: URL:https://arxiv.org/pdf/2304.08069 (дата обращения: 12.01.25).

9. Roboflow [Электронный ресурс]: URL:https://universe.roboflow.com/alisa-petrova/grocery-items-9rtd2/dataset/2 (дата обращения: 17.12.24).

10. Datasets [Электронный ресурс]: URL:https://blog.roboflow.com/smart-fridge/ (дата обращения: 17.12.24).

11. Datasets [Электронный ресурс]: URL:http://foodcam.mobi/dataset.html (дата обращения: 17.12.24).

519.175.3

Пономарев Ю. Ю.

Вопросы точных оценок характеристик деревьев при моделировании сверхбольших графов технологических и естественных сетей с заданными статистиками вершин, ребер и прохождения сигналов

Введение. Моделирование информационных систем порождает графовые отображения со сверхбольшим количеством вершин. Ранее в статьях [1,2], было показано, как термы определенной глубины вложенности могут быть описаны универсальным объектом, называемым DOWN-UP код. Общий анализ сверхбольших графов технологических и естественных сетей в обычных постановках задач перечисления, построения и синтеза как правило приводит к заключению, что современные языки не способны оперировать числами с большим количеством нулей.

В ответ на этот вызов, комитеты по стандартизации ряда известных языков заранее приняли доктрины последовательного наращивания возможностей развиваемых ими языков в сторону создания библиотек для работы с числами с бесконечной точностью. В некоторых языках (Java, Python) эти проблемы были решены изначально для криптографии, либо как расширение самого языка. Все это необходимо, чтобы реализовать вычисление по формулам, основанным на свойствах кода Прюфера.

В работе [3] автор развивает другой подход к задачам синтеза любой иерархической структуры, так как очень часто нужна модель, построенная на статистиках вершин, ребер, оценках задержки прохождения сигнала по ребрам древовидного графа.

Классические результаты теории графов описаны в [4], однако заметим, что моделированием графов при числе ребер свыше 100000

Пономарев Юрий Юрьевич – эксперт, ООО «Автоматизированные системы, консалтинг, информационные технологии»; e-mail: yura_graph@mail.ru, тел.: +7(903)537-46-11

занимаются больше специалисты по теории вероятностей, а не специалисты по графам.

Ранее в [2] уже были приведены ряд теорем о DOWN-UP кодах, но без отдельного анализа их эффективности при расчетах с их помощью на заданном объеме оперативной памяти, длины кода, максимальном и минимальном элементах.

Для конкретизации напомним определение DOWN-UP кодов, которые определяются рекурсивно.

Далее для простоты изложения положим $A = a_1$, $a_1 = 1$ либо $A = 1$ – граничное условие на код, единичный код. В общем виде код есть последовательность из элементов с четными и нечетными индексами: $A = \{a_{2i-1} \in N,\, a_{2i} \in N \cup \{0\},\, i \in N\}$. При этом элементы, имеющие четные индексы, строго меньше обоих соседних с ними элементов с нечетными индексами, $|A|$ – число элементов в коде или размерность кода, число всегда нечетное, $|A| \geqslant 3$.

Как показано в [5], существует биекция между множеством корневых деревьев и DOWN-UP кодов, что позволяет распространять результаты, полученные для DOWN-UP кодов, на все множество упорядоченных корневых деревьев без потери общности сформулированных в статье утверждений. Классическая теорема Кэли о числе корневых деревьев дает n^{n-2} кодов Прюфера для корневых деревьев с числом вершин n.

В статьях [2, 6], представлено несколько теорем и следствий из них, которые утверждают, что многие особенности DOWN-UP кода могут быть установлены в виде описания его элементов.

Описание предметной области. Далее будем следовать определению понятия предметной области, предложенному в [7] и [2].

Предметная область **D** спецификации $(c_0^0, {c_1}^{m_1}, \ldots, c_n^{m_k})$ – это совокупность термов, получаемых из данного количества $c_0 \neq 0$ знаков переменных $x_1, \ldots, x_{c_0}$ и данных количеств c_i m_i-местных функциональных знаков

$$f_1^{(m_i)}, \ldots, {f_{c_i}}^{(m_i)}, \quad i = 1, \ldots, n,$$

соединяемых с помощью композиции.

Терм, как обычно, определяется индуктивно: 1) переменная есть терм, 2) если $f_{(i)}$ – i-местный функциональный знак и $a_1, \ldots, a_i$ – термы, то $f_{(i)}(a_1, \ldots, a_i)$ – терм, 3) никаких других термов не существует.

Глубиной вложенности данного вхождения терма $a \in \mathbf{D}$, как подтерма в терм $b \in \mathbf{D}$, назовем число термов, содержащих данное вхождение. Максимальную из глубин вложенности всех вхождений переменных в терм назовем глубиной терма. На языке функциональных деревьев, представляющих структуру терма, глубина терма – высота соответствующего функционального дерева.

Постановка задачи. Для корневых деревьев создать теоремы-аналоги, ранее сформулированные различными исследователями для кодов Прюфера. Программными методами сравнить результаты и оценить их эффективность на сверхбольших графах. По-возможности, учесть специфику задач синтеза при статистических исследованиях графов и наработки статистики для сравнения с существующими вероятностными подходами оценивания параметров деревьев и прохождения сигналов по ним.

Основные результаты. Приведем теоремы, показывающие как предельные свойства DOWN-UP кода, так и возможную потребность в ресурсах ЭВМ при работе со сверхбольшими деревьями.

Теорема 1. *Существует и для каждого дерева (корневого и некорневого) единственна система из DOWN-UP кода и вектора номеров вершин, аналогичная его коду Прюфера.*

Доказательство данной теоремы вытекает из свойств уровневого кода, получаемого из заданного DOWN-UP кода, описанного в [8]. Если же дерево некорневое, то требуется помечать как корень каждую некорневую вершину и получать систему из DOWN-UP кодов.

Теорема 2. *Размерности DOWN-UP кодов, подчиняющиеся ранее выведенным теоремам, позволяют эффективно предсказывать сложность задачи анализа сверхбольшого дерева.*

Доказательство основывается на теореме из статьи [2] о максимальном числе DOWN-UP кодов для заданного числа вершин в корневом дереве и размерности кода.

Объемы памяти ЭВМ для обсчета графов на популярных языках программирования в случае 4-байтного кодирования каждого элемента кода приведены ниже.

Теорема 3. *Для статистик вершин в сверхбольших деревьях можно найти вероятностные оценки, аналогичные коду Прюфера, но за меньшее время.*

Доказательство. Структурные элементы корневого дерева: корень, концевая вершина, внутренняя вершина и точки ветвления

полностью идентифицированы в работе [1]. Предельные теоремы об их числе ничем по структуре доказательства не отличаются от теоремы из работы [8] о числе вершин в корневом дереве, сопоставленным кодом дерева с заданным числом вершин и размерностью кода.

Несколько утверждений о потреблении памяти ЭВМ при реализации анализа дерева на различных языках программирования.

Теорема 4. *Прохождение сигналов по сверхбольшому дереву (корневому и некорневому) определяется в большинстве случаев по среднему значению статистик для заданного множества графов с заданным числом концевых вершин за линейное время.*

Данная теорема имеет большое количество приложений. Поэтому приведем иллюстрацию ее сложности в коде на языке Python 3.7.

```
from functools import lru_cache
from sympy import divisors
import math
@lru_cache(maxsize=None)
def divisor_tuple(n): # cached unordered tuple of divisors
    return tuple(divisors(n, generator=True))
@lru_cache(maxsize=None)
def A000081(n): return n if n <= 1 else
 sum(sum(d*A000081(d)
 for d in divisor_tuple(k))*A000081(n-k)
 for k in range(1, n))//(n-1) # Chai Wah Wu, Jan 14 2022
def A000055(n): return 1 if n == 0 else
A000081(n)-sum(A000081(i)*A000081(n-i)
for i in range(1, n//2+1)) +
(0 if n % 2 else (A000081(n//2)+1)*A000081(n//2)//2)
a = 0
for i in range(3,600):
      print (i," ",A000055(i)," ",A000081(i), " ",
      math.ceil(a*(3.0/1.015))," ",
      math.ceil(a*(3.0/1.015))/A000081(i)," ",
      math.ceil((0.44935632*math.pow(2.95576,i*1.0))/
      (i**1.5))/A000081(i))
      a = A000081(i)
```

Вычисление массивов A000055, A000081 взято из ресурсов [9, 10].

Об эффективности кодирования корневых деревьев DOWN-UP кодами говорит следующая

Теорема 5. *Сравнение эффективности кодирования кодом Прюфера и DOWN-UP кодом показывает отношение длины кодов в пределе по числу вершин, равное 1.*

Доказательство. Эмпирическое число кодов для заданного числа вершин n есть $\frac{\binom{2(n-1)}{n-1}-n}{n}$. К каждому коду нужно учесть $n!$ его изоморфных состояний. Определим, что кодирование корневого дерева будет оптимальным, если $\lim\limits_{n\to\infty}\frac{j(n)}{H(n)}=1$. Для кода Прюфера данное соотношение выполняется по определению: $\lim\limits_{n\to\infty}\frac{(n-2)\log(n)}{(n-2)\log(n)}=1$, где под log подразумевается $\log_2$. Используем две асимтотики для удобства нахождения логарифма при вычислении $j(n)=\log(\sqrt{2n\pi}(\frac{n}{\exp})^n\frac{\frac{2^{2n}}{\sqrt{n\pi}}-n}{n})$ в нашем случае. Раскрывая скобки в $j(n)$, производя приведение подобных членов и находя предел отношения, получаем 1.

Рассмотрим прием сокращения памяти ЭВМ для обработки DOWN-UP кода в предыдущей теореме с использованием соображений симметрии.

Теорема 6. *Если принять простейший способ декодирования дерева по правилу «код декодируется по структурным элементам сверху вниз и слева направо по всем структурным элементам», то предел, указанный в теореме 5, может быть меньше 1.*

Доказательство основывается на свойствах симметричности кода, выведенных в статье [2].

Практические наблюдения за возможностью анализа кода дают следующие теоремы.

Теорема 7. *На языке Java максимальный размер массива есть* $d_{\max}=2147483647$ *элементов, значит максимальное число вершин* $n=\lceil(d_{\max}+3)/2\rceil$.

Доказательство. Значение для заданного числа вершин дерева n найдено на основании формулы максимальной размерности DOWN-UP кода.

Для практических вычислений необходимо иметь запас памяти ЭВМ для сохранения определенного числа уровней кода для анализа. Тест для размещения кода в оперативной памяти для языка C:

```
#define N 1000000000
static int list[N];
int main(int argc, char **argv)
```

```
{
size_t i;
srand(time(NULL));
for(i=0; i<N; i++)
list[i] = rand()%1000;
return 0;
}
```

Данный тест потребует примерно 4 ГБ оперативной памяти.

Тест для размещения кода в оперативной памяти для языка Python:

```
import numpy as np
np.arrange(15000000000,dtype=np.type('int32'))
```

Данный тест потребует примерно 57 ГБ оперативной памяти. Заметим, что вызов numpy позволяет больше понять возможности компьютера.

Вероятность ветвления корневого дерева по всем возможным кодам для заданного числа вершин без учета изоморфизма фиксирует

Теорема 8. *Вероятность появления ветвящегося дерева среди всех возможных кодов для заданного числа вершин n составляет* $1 - 2^{-2n}$.

Доказательство. Доопределим DOWN-UP код в виде одномерного массива (n). Имеем ровно $n!$ таких изоморфных неветвящихся деревьев. Так как число кодов для ветвящихся корневых деревьев есть $n!\frac{\binom{2(n-1)}{n-1}-n}{n}$ с учетом всех изоморфных деревьев, то $n!\frac{\binom{2(n-1)}{n-1}-n}{n} + n!$ есть общее число деревьев. Находим предел для вероятности: $1 - \lim\limits_{n\to\infty} \frac{n!}{n!+n!\frac{\binom{2(n-1)}{n-1}-n}{n}} = 1 - 2^{-2n}$.

Заключение. Результаты данной работы обсуждались на семинарах 57-й лаборатории ИПУ РАН. Они апробированы на моделях организационных систем в качестве предельных теорем [3].

Литература

1. Пономарев Ю. Ю. Кодирование корневых деревьев DOWN-UP последовательностями // Процессы управления и устойчивость: Труды 29-й научной конференции / под ред. В. Н. Старкова. СПб.: НИИ Химии СПбГУ, 1998. С. 257–259.

2. Пономарев Ю. Ю. Вопросы точных оценок характеристик деревьев при моделировании термов предметных областей, нейросетей, XML данных // Процессы управления и устойчивость. 2023. Т. 10. № 1. С. 309–317.

3. Пономарев Ю. Ю. Точные оценки числа решений задачи синтеза иерархических систем управления в постановке М.И. Рубинштейна // Теория активных систем – 55 лет (ТАС-55): труды юбилейной научно-практической конференции / под общей редакцией Д. А. Новикова, В. Н. Буркова. М.: ИПУ РАН, 2024. С. 61–66.

4. Харари Ф., Палмер Э. Перечисление графов / пер. с англ. Г. П. Гаврилова. М.: Мир, 1977. 324 с.

5. Волошин Ю. М. Последовательности «ап-даун» и упорядоченные деревья // Доклады РАН. 1993. Т. 329. № 2. С. 125–127.

6. Пономарев Ю. Ю. Вопросы точных оценок характеристик деревьев при моделировании метаданных в информационных системах общего назначения // Процессы управления и устойчивость. 2024. Т. 11. № 1. С. 253–261.

7. Волошин Ю. М. Перечисление термов предметных областей по глубине вложенности // Доклады АН СССР. 1974. Т. 219. № 6. С. 1294–1297.

8. Пономарев Ю. Ю. DOWN-UP последовательности и корневые деревья // Процессы управления и устойчивость: Труды 30-й научной конференции / под ред. В. Н. Старкова. СПб.: НИИ Химии СПбГУ, 1999. С. 374–380.

9. Онлайн-энциклопедия целочисленных последовательностей [Электронный ресурс]: URL:https://oeis.org/search?q=+A000055&language=russian (дата обращения: 01.04.2025).

10. Онлайн-энциклопедия целочисленных последовательностей [Электронный ресурс]: URL:https://oeis.org/search?q=+A000081&language=russian (дата обращения: 01.04.2025).

УДК 004.8

Попов А. П.

Методы построения интеллектуальных ассистентов на основе LLM для обработки открытых медицинских данных

Рекомендовано к публикации доцентом Блекановым И. С.

1. Актуальность работы. Современная система здравоохранения претерпевает существенные изменения ввиду внедрения нейросетевых технологий для диагностики заболеваний [1]. Однако медицинские организации всё ещё сталкиваются с растущими требованиями к качеству и скорости оказания услуг. Пациенты ожидают оперативного получения информации о доступных им процедурах, врачах и новостях медицинских учреждений. Обработка запросов сотрудниками регистратуры часто оказывается трудоёмкой и не эффективной в условиях высокой загруженности персонала.

Согласно исследованиям [2] внедрение чат-ботов и виртуальных ассистентов в здравоохранение может позволить не только сократить время обработки запросов пациентов, но и повысить их уровень удовлетворенности от взаимодействия с медицинскими учреждениями. Такие системы особенно полезны в условиях чрезвычайных ситуаций, когда спрос на дистанционное оказание услуг увеличивается.

В то же время стремительно развивающиеся в последнее десятилетие большие языковые модели позволяют решать всё более широкий спектр задач в области обработки естественного языка [3]. В частности, одной из популярных задач является задача построения вопросно-ответных систем (Question Answering) [4], нашедшая своё применение в различных областях.

В данной работе рассматривается построение интеллектуального ассистента на основе RAG подхода [5] для повышения информированности пациентов медицинской организации и уровня цифровизации системы здравоохранения с использованием специализированного набора данных и приводится оценка качества полученных результатов.

Попов Артём Петрович – магистрант, Санкт-Петербургский государственный университет; e-mail: st076324@student.spbu.ru, тел.: +7(913)053-36-89

2. Постановка задачи. Рассмотрим информационный запрос пользователя Q, ответ на запрос A_{th} и источник ответа C_{th}. Необходимо разработать интеллектуальную систему на основе открытых данных сайта больницы, которая способна автоматически обрабатывать запрос пользователя, генерируя ответ A_{gen} и предоставляя список релевантных запросу источников C_{rel} за время t. Для построения вопросно-ответных систем могут применяться различные техники дообучения LLM, такие как LoRA, однако RAG подход, как правило, показывает более высокие оценки качества и предоставляет гибкий доступ к валидации результатов модели [6].

Таким образом, задача сводится к минимизации ошибок генерации ответа $\mathcal{L}_{ans}(A_{th}, A_{gen})$ и поиска релевантных источников $\mathcal{L}_r(C_{th}, C_{rel})$ в контексте некоторых специализированных метрик рассматриваемой задачи и RAG подхода.

3. Предлагаемое решение. Пайплайн системы состоит из четырёх основных модулей, представленных на рис. 1. Первые три модуля позволяют сформировать базу знаний медицинского учреждения, которая используется на этапе подготовки ответа на вопрос.

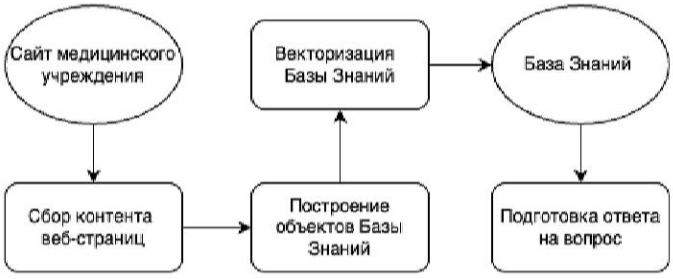


Рис. 1. Общая архитектура решения

3.1. Сбор контента веб-страниц. В качестве источника данных для базы знаний был выбран сайт Клиники высоких медицинских технологий им. Н. И. Пирогова. Для сбора данных с сайта больницы был разработан веб-краулер на основе фреймворка Scrapy, который позволяет эффективно обходить веб-страницы и извлекать из них необходимую информацию. Всего было собрано более 2000 веб-страниц сайта. Для каждой веб-страницы помимо парсинга основного контента собирались заголовок и дата публикации.

3.2. Построение объектов базы знаний. Поскольку контекстное окно больших языковых моделей ограничено, а контент в рамках даже одной страницы может содержать разные тематики, для повышения качества поиска и ответов моделей производится разбиение данных на чанки (фрагменты текста). Это также позволяет снизить риск галлюцинаций модели.

Существуют различные подходы к формированию чанков исходя из определённых предположений о базе знаний [7], в частности, в данной работе чанки имеют динамический размер, определяющийся синтаксисом текста. Для повышения устойчивости и информативности каждый чанк содержит метаинформацию с полей, упомянутых в пункте 3.1.

3.3. Векторизация базы знаний. Для обеспечения быстрого и более полного поиска информации в медицинском ассистенте используется ансамблевый подход, сочетающий такие методы, как BM25 и векторизацию на основе эмбеддинговой модели e5-base multilingual [8]. Формула BM25 имеет следующий вид:

$$\text{BM25}(D, Q) = \sum_{i=1}^{n} \text{IDF}(q_i) \frac{f(q_i, D)(k_1 + 1)}{f(q_i, D) + k_1 \left(1 - b + b \frac{|D|}{\text{avgdl}}\right)},$$

где D – документ, $Q = \{q_1, \ldots, q_n\}$ – запрос, состоящий из термов q_i, $f(q_i, D)$ – частота терма q_i в документе D, $|D|$ – длина документа D (количество слов), avgdl – средняя длина документа в коллекции, k_1 и b – параметры, обычно $k_1 \in [1{,}2; 2{,}0]$ и $b = 0{,}75$, $\text{IDF}(q_i)$ – обратная документная частота терма q_i

Метод BM25 используется для осуществления полнотекстового поиска на основе ключевых слов. Это требует дополнительной предобратки текста в виде нормализации (стемминга или лемматизации) для сжатия словаря поиска. В то же время эмбеддинговая модель проводит поиск на основе сематики запроса, позволяя использовать синонимичные слова и фразы.

3.4. Подготовка ответа на вопрос. В качестве большой языковой модели в решении используется API российской модели GigaChat Max с 32 тысячами токенов в контекстном окне. Пайплайн подготовки ответа на вопрос включает в себя несколько этапов, представленных на рис. 2.

Дополнение поискового запроса на основе истории диалога с пользователем позволяет при необходимости задавать серию уточняющих вопросов. Далее осуществляется векторизация поискового запроса с помощью той же модели, что и чанки в базе знаний. Кроме того, поиск релевантных источников должен учитывать ограничение на максимальный размер контекста. Финальным этапом является генерация ответа с помощью LLM с учетом описанной роли и ограничений для ассистента в системном промпте.



Рис. 2. Пайплайн подготовки ответа на вопрос

4. Эксперимент. Для оценки работы вопросно-ответных систем выделяют два основных критерия – качество поиска релевантных источников и качество сгенерированного ответа. Наиболее точной является их ручная оценка с помощью разметки с перекрытием, однако также широко используются и автомазированные подходы.

4.1. Постановка эксперимента. Одним из наиболее популярных фреймворков для автоматизации оценки качества является RAGAS [9]. Однако подобные инструменты производят оценку как правило с помощью LLM-оценщика и выбранной эмбеддинговой модели, что вносит свои ограничения в объективность результата.

Дополнительно в работе рассматриваются две специализированные метрики. Первая метрика оценивает качество поиска по бинарной шкале, определяющей, была ли найдена релевантная ссылка поисковой системой. Вторая оценивает качество генерации ответа с точки зрения фактологической точности по пятибальной шкале.

Для проведения оценки качества был подготовлен тестовый сет

из 60 вопросов, покрывающий основные тематические блоки сайта. Пример одного из вопросов представлен в таблице 1.

Таблица 1. Пример тестового вопроса

Поле	**Значение**
Вопрос	Какие документы нужны для оформления талона для вмп?
Тематика	Организационное
Ответ	1. Направление на госпитализацию для оказания ВМП; <...> 6. Согласие пациента на обработку персональных данных
Источник	https://www.gosmed.ru/vmp/protsedura-polucheniya-vmp-grazhdanami-rf.php

4.2. Результат эксперимента. Значения метрик фреймворка RAGAS для тестового сета представлены в таблице 2.

Таблица 2. Значения метрик RAGAS

Название метрики	**Значение**
factual correctness	0,333
faithfulness	0,736
context precision	0,472
context recall	0,588

Более объективными и интерпретируемыми являются результаты специализированных метрик. Доля верно найденных релевантных источников составила 0,65. Распределение ответов по пятибальной шкале представлено в таблице 3.

Таблица 3. Оценка качества генерации ответа

Оценка	5	4	3	2	1
Доля ответов	0,38	0,18	0,23	0,11	0,10

5. Заключение. Таким образом, реализован автоматизированный сбор и обработка данных, а на собранном датасете построен медицинский ассистент на основе LLM, показывающий хорошую точность работы. Универсальная архитектура системы позволяет легко масштабировать решение на другие медицинские организации. В дальнейшем предполагается расширение функциональности ассистента в контексте помощи врачам при постановке диагноза пациенту и создание AI-агента на его основе.

Литература

1. Ляхова У. А., Ляхов П. А. Мультимодальная ансамблевая нейросетевая система обнаружения рака кожи на генерогенных дерматологических данных // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2024. Т. 20. № 2. С. 231–243.

2. Clark M., Bailey S. Chatbots in health care: connecting patients to information // Canadian Agency for Drugs and Technologies in Health. 2024. Vol. 4. Iss. 1. P. 1–22.

3. Qin L., Chen Q., Feng X. et al. Large language models meet NLP: a survey [Электронный ресурс]: URL:https://arxiv.org/abs/2405.12819 (дата обращения: 09.03.25).

4. Pandya H. A., Bhatt B. S. Question answering survey: directions, challenges, datasets, evaluation matrices [Электронный ресурс]: URL:https://arxiv.org/abs/2112.03572 (дата обращения: 10.03.25)

5. Qin L., Chen Q., Feng X. et al. Retrieval-augmented generation for large language models: a survey [Электронный ресурс]: URL:https://arxiv.org/abs/2312.10997 (дата обращения: 09.03.25).

6. Павлов А. В. Методы точной настройки: RAG VS LORA // Процессы управления и устойчивость. 2024. Т. 11. № 1. С. 241–246.

7. Kshirsagar A. Enhancing RAG performance through chunking and text splitting techniques // International Journal of Scientific Research in Computer Science Engineering and Information Technology. 2024. Vol. 10. Iss. 5. P. 151–158.

8. Wang L., Yang N., Huang X. Text embeddings by weakly-supervised contrastive pre-training [Электронный ресурс]: URL:https://arxiv.org/abs/2212.03533 (дата обращения: 10.03.25).

9. Es S., James J., Espinosa-Anke L. et al. RAGAS: automated evaluation of retrieval augmented generation [Электронный ресурс]: URL:https://arxiv.org/abs/2309.15217 (дата обращения: 09.03.25).

УДК 519.234.3

Похабов И. И., Гориховский В. И.

Графовые критерии проверки гипотез согласия

1. Введение. Математическая статистика применяется практически во всех областях, например, при проверке биологических, медицинских и прочих гипотез [1,2]. Одной из главных задач математической статистики является проверка гипотез согласия с распределениями. Критерием согласия называют функцию, которая по выборке и уровню значимости позволяет отвергнуть гипотезу о принадлежности выборки к заданному распределению. Несмотря на обилие различных критериев, как общих (критерий Колмогорова – Смирнова или критерий хи-квадрат), так и специализированных (критерий Шапиро – Уилка), не существует универсальных критериев, позволяющих уверенно принимать гипотезы. Развитие новых критериев позволяет повышать уверенность принятия и отвержения гипотез. В представленной работе рассматривается возможность использования графовых численных характеристик выборки для получения новых критериев согласия.

Графовый подход к проверке гипотез уже применяется, например, для решения задачи обнаружения разладки [3]. В этой работе рассматриваются дистанционные графы, построенные по элементам выборки, на основе которых строится комбинаторный критерий наличия разладки. Этот подход может быть обобщен для проверки гипотез согласия и является достаточно перспективным, поскольку использует структурные характеристики выборки, фактически не рассматриваемые другими критериями.

В данной статье исследуется корреляция между простейшими численными характеристиками графов выборок и принадлежностью к семействам распределений, а также представлены результа-

Похабов Иван Ильич – студент, Санкт-Петербургский государственный университет; e-mail: ivan.pokhabov.23@gmail.com, тел.: +7(913)190-25-11

Гориховский Вячеслав Игоревич – доцент, Санкт-Петербургский государственный университет; e-mail: gorihovskyvyacheslav@gmail.com, тел.: +7(921)344-41-97

Работа выполнена при финансовой поддержке СПбГУ (Pure ID 116636233)

ты оценки мощности некоторых графовых критериев нормальности методом Монте-Карло моделирования.

2. Критерии согласия. Среди всех критериев согласия можно выделить два основных класса:

- *параметрические* (предполагают согласие с распределением из известного семейства распределений и известными параметрами). Например, критерии хи-квадрат [4] и Шапиро – Уилка [5]. Недостатками критериев являются требование соответствия распределению и чувствительность к параметрам;
- *непараметрические* (предполагают согласие с распределением из известного семейства распределений, но не требует информации о параметрах). Например, критерии Манна – Уитни [6] и Колмогорова – Смирнова [7, 8]. Недостатками критериев являются меньшая мощность, ограниченная интерпретация.

Также встречаются семипараметрические критерии, для которых известна лишь часть параметров.

2.1. Графовые критерии согласия. Графовые критерии устроены следующим образом: критерий вычисляет параметр дистанции для выборки и строит граф, в котором вершинами являются элементы выборки. Ребро между вершинами строится в том случае, если абсолютная разность между значениями элементов выборки в них меньше значения параметра дистанции. После этого вычисляются численные характеристики построенного графа и сравниваются с эталонными значениями или проводится моделирование методом Монте-Карло, после чего гипотеза либо отвергается, либо принимается. Графовые критерии могут быть как параметрическими, так и непараметрическими. Для этого критерий в начале нормирует данные или учитывает параметры выборки при вычислении параметра дистанции.

В данной работе сравниваются значения статистик на следующих распределениях: *нормальное, экспоненциальное и равномерное.*

Для изучения влияния различных распределений на дистанционные графы были выбраны следующие характеристики: *количество ребер, максимальная степень вершины, количество компонент связности.*

После проведения ряда измерений было принято решение отложить изучение количества компонент связности, так как не был по-

добран параметр дистанции, который бы показывал явную корреляцию между данными и выбранным распределением.

Были вычислены средние значения выбранных характеристик для выборок размером 100, 150, ..., 2000, используя выбранные распределения со случайными параметрами. Графики количества ребер и максимальной степени вершины представлены на рис. 1 и 2. Для построения графов был выбран параметр дистанции

$$\text{dist} = \frac{\max(S) - \min(S)}{10},$$

где S – выборка. Параметр константы был выбран так, чтобы количество ребер в графе для маленьких выборок было более отличимым, а для больших не требовало слишком долгих вычислений.

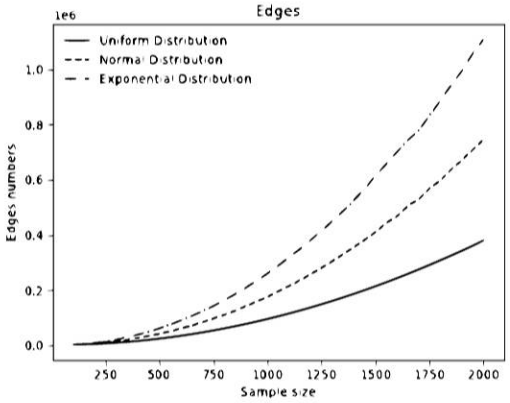


Рис. 1. График количества ребер

На графиках видно, что данные функции явно зависят от семейства распределения и могут быть полезны для графовых критериев согласия. График на рис. 1 позволяет сказать, что эта характеристика может быть представлена как квадратичная функция относительно размера выборки, а на рис. 2 – как линейная.

3. Экспериментальное исследование критериев. Были реализованы графовые критерии нормальности, основанные на подсчете ребер и максимального значения степени вершины. Реализация была выполнена в рамках развивающегося проекта *PySATL* [9], с

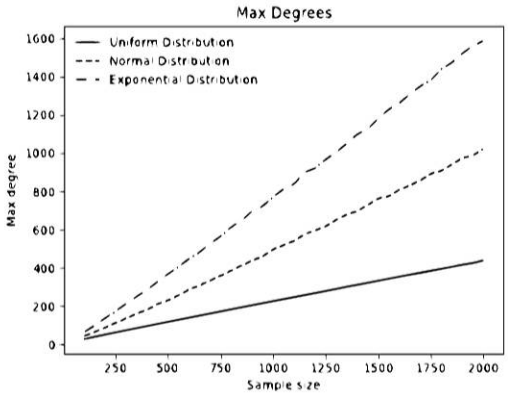


Рис. 2. График максимальной степени вершины

которой можно ознакомиться по ссылке в сносках. На данный момент предельные распределения статистик не были найдены, поэтому критические значения находятся с помощью метода Монте-Карло-моделирования, а параметр дистанции вычисляется по формуле

$$\text{dist} = \frac{\max(S) - \min(S)}{10\sigma^2},$$

где S – выборка, а σ^2 – ее дисперсия.

Также была проведена оценка мощности критериев и сравнение с критерием Колмогорова – Смирнова с помощью встроенных в библиотеку проекта инструментов. Параметры эксперимента:

- в качестве нулевой гипотезы использовались два нормальных распределения $\mathcal{N}(0, 1)$ и $\mathcal{N}(10, 20)$;
- уровень значимости: $\alpha = 0{,}05$;
- вычисление критических значений было двусторонним, использовалось моделирование методом Монте-Карло с количеством итераций, равным 1000000;
- в качестве альтернатив были выбраны следующие распределения: гамма ($\alpha = 1$, $\beta = 2$), гамма ($\alpha = 0{,}5$, $\beta = 1$), экспоненциальное ($\lambda = 2$), Вейбулла ($a = 1$, $k = 2$), хи-квадрат ($df = 5$), Лапласа ($t = 0$, $s = 1$);
- количество альтернативных выборок равно 100, размер выборок: $10, 20, 30, 40, 50$.

С результатами эксперимента можно ознакомиться в [9, 10].

3.1. Основные результаты. Получены следующие результаты:

- EdgesNumber чувствителен к различным альтернативным распределениям, особенно при увеличении выборки;
- MaxDegree наиболее мощен для распределений Вейбулла, гамма и экспоненциального на малых выборках;
- при $n = 10$ и $n = 20$ новые критерии умеренно эффективны, при $n = 50$ их мощность приближается к единице;
- в зависимости от альтернатив новые критерии могут показывать результаты, лучшие, чем уже устоявшийся критерий Колмогорова – Смирнова (распределение Лапласа при нулевой гипотезе $\mathcal{N}(10, 20)$);
- на большей части альтернатив при небольшом размере выборки новые критерии уступают критерию Колмогорова – Смирнова.

4. Заключение. Был предложен новый подход для формирования критериев согласия, основанный на вычислении численных характеристик дистанционных графов. Реализованные критерии нормальности, основанные на подсчете количества ребер и максимальной степени вершины в графе, показали многообещающие результаты, но требуют дальнейших исследований и новых реализаций.

Литература

1. Motulsky H. J. Statistical concepts in biology and health sciences [Электронный ресурс] // ResearchGate. 2014. `https://www.researchgate.net/publication/266021540_Statistical_concepts_in_biology_and_health_sciences` (дата обращения: 28.02.2025).

2. Wasserstein R. L., Schirm A. L., Lazar N. A. Moving to a world beyond "p < 0.05" // The American Statistician. 2022. Vol. 76. No 1. P. 1–19.

3. Chen H., Zhang N. Graph-based change-point detection // Annals of Statistics. 2015. Vol. 43. No 1. P. 139–176.

4. Pearson K. On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling // Philosophical Magazine. 1900. Vol. 50. No 302. P. 157–175.

5. Shapiro S. S., Wilk M. B. An analysis of variance test for normality (complete samples) // Biometrika. 1965. Vol. 52. No 3-4. P. 591–611.

6. Mann H. B., Whitney D. R. On a test of whether one of two random variables is stochastically larger than the other // The Annals of Mathematical Statistics. 1947. Vol. 18. No 1. P. 50–60.

7. Kolmogorov A. N. Sulla determinazione empirica di una legge di distribuzione // Giornale dell'Istituto Italiano degli Attuari. 1933. Vol. 4. P. 83–91.

8. Smirnov N. V. On the estimation of the discrepancy between empirical curves of distribution for two independent samples // Bulletin de l'Universite de Moscou. 1939. Vol. 2. No 2. P. 3–16.

9. Graph-based normality tests [Электронный ресурс]: URL:https://github.com/PySATL/pysatl-criterion/blob/main/pysatl_criterion/normal.py (дата обращения: 13.04.25).

10. Power experiment results [Электронный ресурс]: URL:https://github.com/Ivan-Pokhabov/pysatl-experiment/blob/main/graph_norm_results.pdf (дата обращения: 13.04.25).

УДК 004.852

Саперов Д. П.

Оценка влияния мировых событий на рынок криптовалют

Рекомендовано к публикации профессором Крылатовым А. Ю.

1. Введение. С момента появления биткоина в 2009 году криптовалюты стали неотъемлемым элементом глобальной финансовой системы. Одной из ключевых характеристик рынка криптовалют является высокая волатильность, что делает его привлекательным для инвесторов, но одновременно повышает уровень рисков. Фактором, оказывающим значительное влияние на динамику цен криптовалют, являются мировые события, включая экономические кризисы, изменения в законодательстве, технологические инновации, а также новости, связанные с крупнейшими игроками криптоиндустрии.

2. Постановка задачи. В рамках исследования разработана модель для оценки влияния мировых событий и новостей на цену биткоина. Цель – создать точные инструменты анализа, которые учтут как временные тренды, так и влияние новостей на цену.

Предполагается использование гибридного подхода, включающего модель LSTM для анализа временных рядов и методы градиентного бустинга и случайного леса для оценки воздействия новостных событий. Ключевым этапом работы является создание системы классификации новостных заголовков, для этого будет применяться zero-shot классификация с моделью BERT, что позволит автоматически распределять новости по более узким категориям.

Оценка качества модели будет проводиться с помощью метрик MAE и RMSE.

3. Сбор и обработка данных. Для проведения исследования использовались данные по историческим ценам биткоина, которые включают поминутные котировки с параметрами объема торгов, цен открытия, закрытия и других характеристик. Эти данные были получены из открытых источников [1]. В дополнение к этим данным

Саперов Дмитрий Петрович – студент, Санкт-Петербургский государственный университет; e-mail: dima.saperov.03@gmail.com, тел.: +7(912)329-14-94

были рассчитаны различные технические индикаторы, такие как индекс относительной силы (RSI), скользящие средние (SMA), процентное изменение цены актива за определенный период времени, волатильность, а также день недели и час дня для выявления сезонных изменений в ценовой динамике. Эти параметры важны для анализа временных трендов и предсказания будущих изменений цен.

Для корректной работы модели данные были нормализованы. Процесс нормализации данных включает в себя приведение их к единой шкале, чтобы различные признаки имели одинаковое влияние на модель. Математически нормализация данных может быть выражена следующим образом:

$$X_{\text{norm}} = \frac{X - \mu}{\sigma},$$

где X – это значение исходного признака, μ – среднее значение признака, а σ – стандартное отклонение признака. Такой подход, называемый стандартизацией, приводит данные к нулевому среднему и единичному стандартному отклонению, что помогает улучшить сходимость модели и повышает ее стабильность.

Помимо ценовых данных, были получены новостные заголовки с временными метками и категориями [2]. Их временные метки были синхронизированы с историческими данными о ценах биткоина для учета влияния новостей на динамику цен.

Исходные категории новостей, такие как «Мир», «Экономика» и «Наука и техника» и другие, оказались слишком широкими и не отражали достаточно специфично влияние отдельных событий на цену биткоина. Поэтому было принято решение переклассифицировать их в более узкие и специфичные категории. Для этого можно применить методы выделения семантически близких групп, описанные в статьях [3, 4].

К новостным заголовкам была применена модель word2vec для определения контекстно близких слов, что позволило выявить слова, наиболее связанные с каждой из исходных категорий. На основе этих контекстно близких слов составлены новые, более узкие и специфичные категории с ярко выраженной эмоциональной окраской.

3.1. Zero-shot классификация и модель BERT. Для классификации новостей использован метод zero-shot, который позволяет классифицировать тексты без необходимости дополнительного обучения модели на конкретном наборе данных. Этот метод основыва-

ется на мощных языковых моделях, таких как BERT (Bidirectional Encoder Representations from Transformers). В работе использована модель "Multilingual mDeBERTa-v3-base-mnli-xnli" [5].

Математическая zero-shot классификация работает следующим образом. Пусть θ – это модель, обученная на задаче предсказания вероятности для каждого возможного класса C_k. Для каждого новостного заголовка T модель генерирует вероятности принадлежности этого текста к каждому из возможных классов $C_1, \ldots, C_n$ на основе контекста, извлеченного из текста.

$$P(T|C_k) = \frac{\exp(f_\theta(T, C_k))}{\sum_{i=1}^{n} \exp(f_\theta(T, C_i))},$$

где $f_\theta(T, C_k)$ – это функция, оценивающая связь между текстом T и классом C_k, а exp – экспоненциальная функция для нормализации вероятностей.

BERT представляет собой трансформер с двусторонним контекстом, который учитывает как предшествующие, так и последующие слова в тексте. Это позволяет модели эффективно обрабатывать сложные зависимости и контексты в тексте. Математически BERT обучается с использованием маскированного языкового моделирования, где для каждого слова w_i в предложении из маскируемой последовательности выбирается вероятностное распределение

$$P(w_i|\mathbf{w}_{-i}) = \text{softmax}(W_i h_i),$$

где $\mathbf{w}_{-i}$ – контекстные слова без w_i, а h_i – скрытое состояние модели на позиции i, полученное с помощью трансформеров.

В процессе zero-shot классификации модель BERT вычисляет вероятности принадлежности текста к каждой из категорий, и заголовок новостей присваивается категории с наибольшей вероятностью. Это позволяет эффективно классифицировать новости без необходимости дополнительной дообученности на специфических данных.

4. Обучение модели. В процессе обучения модели был использован гибридный подход, который сочетает в себе методы для учета временных зависимостей и влияния новостных событий на цену биткоина.

4.1. LSTM. Для предсказания цен на биткоин с учетом временных зависимостей использована модель LSTM (Long Short-Term

Memory), которая является разновидностью рекуррентных нейронных сетей [6]. Ключевым компонентом данной модели является LSTM-блок, классическая структура которого показана на рис. 1. Архитектура сети включает в себя один LSTM-блок, а также выходной полносвязный слой. Гиперпараметры модели представлены в таблицах 1, 2.

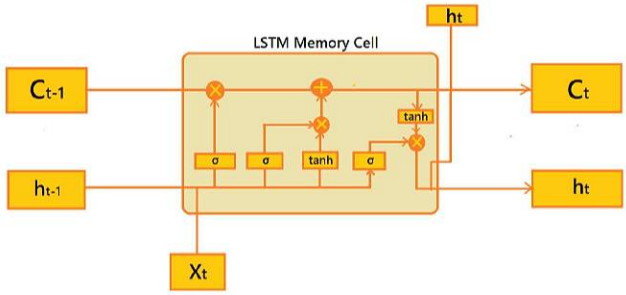


Рис. 1. Структура LSTM-блока

Таблица 1. Параметры модели

Гиперпараметр	
Число LSTM блоков	3
Число нейронов в слоях	50
Число нейронов в выходном слое	1

Таблица 2. Параметры обучения

Гиперпараметр	
Оптимизатор	Adam
Learning rate	0,001
Функция потерь	MSE
Количество эпох	50
Batch size	32

Для реализации модели использовалась библиотека TensorFlow [7].

4.2. Градиентный бустинг (GBM) и случайный лес (RF). После прогнозирования цен с помощью модели LSTM, на полученные данные были применены методы машинного обучения, такие как градиентный бустинг (GBM) и случайный лес (RF). Эти модели были использованы для выявления зависимостей между спрогнозированными ценами и новостными заголовками, которые могут оказывать влияние на рынок.

Градиентный бустинг (GBM) – это метод, который строит ансамбль слабых моделей (деревьев решений) и корректирует ошибки предыдущих деревьев. Математически градиентный бустинг можно записать как

$$f(x) = \sum_{m=1}^{M} \alpha_m h_m(x),$$

где $f(x)$ – итоговое предсказание, $h_m(x)$ – дерево решений, α_m – коэффициент для каждого дерева.

Случайный лес (RF) использует несколько независимых деревьев решений, которые строятся на случайных подмножествах признаков и данных. Прогнозы всех деревьев усредняются для получения итогового результата.

5. Результаты. После обучения модели LSTM, результаты предсказаний изменений цен на биткоин представлены в таблице 3.

Таблица 3. Результаты обучения модели LSTM

MAE	RMSE
0,0044	0,0058

После обучения моделей градиентного бустинга и случайного леса на основе предсказаний LSTM и категорий новостей, были получены результаты, представленные в таблице 4.

Таблица 4. Результаты гибридного подхода

Model	**MAE**	**RMSE**
GBM	0,0032	0,0045
RF	0,0035	0,0047

Модели градиентного бустинга показали более высокую точность в предсказаниях (по метрикам MAE и RMSE) по сравнению со случайным лесом.

Благодаря оценке параметров моделей, были установлены три категории, которые оказывают наибольшее влияние на ценообразование биткоина: криптовалюты и блокчейн; экономические кризисы; инфляция.

6. Заключение. В рамках данного исследования разработана модель для оценки влияния мировых событий на цену биткоина. Использован гибридный подход, сочетавший метод LSTM для анализа

временных рядов и методы градиентного бустинга и случайного леса для анализа влияния новостей. Классификация новостей была выполнена с использованием zero-shot классификации на базе модели BERT. Удалось выявить темы, которые сильнее всего оказывают влияние на цену биткоина.

Литература

1. Котировки и финансовые новости [Электронный ресурс]: `https://ru.investing.com/` (дата обращения: 19.03.25).

2. Новости России и мира сегодня [Электронный ресурс]: `https://lenta.ru/` (дата обращения: 19.03.25).

3. Малютин Е. А., Бугайченко Д. Ю., Мишенин А. Н. Выделение текстовых трендов в социальной сети ОК // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2017. Т. 13. № 3. С. 313–325.

4. Исаев Р. О., Куприянов И. В., Митяй Г. В., Школин А. Ю. Разработка новостного агрегатора с использованием нейросетей // Процессы управления и устойчивость. 2023. Т. 10. № 1. С. 275–280.

5. Multilingual mDeBERTa-v3-base-mnli-xnli [Электронный ресурс]: `https://huggingface.co/MoritzLaurer/mDeBERTa-v3-base-mnli-xnli` (дата обращения: 19.03.25).

6. Shi J., Jain M., Narasimhan G. Time series forecasting (TSF) using various deep learning models [Электронный ресурс]: `https://arxiv.org/abs/2204.11115` (дата обращения: 19.03.25).

7. TensorFlow [Электронный ресурс]: `https://www.tensorflow.org/` (дата обращения: 19.03.25).

УДК 004.912

Смирнов Д. О., Блеканов И. С.

Методы адаптации больших языковых моделей на примере домена LegalTech

1. Введение. Использование больших языковых моделей (LLM) проникает во все отрасли нашей жизни. За последние несколько лет нейросети из сложной научной концепции превратились в инструмент повседневного использования. Практически каждый хоть раз пытался решить поставленную перед ним задачу при помощи языковых помощников.

Основными задачами, которые ставятся перед большими языковыми моделями, являются: обслуживание клиентов через чат-боты, написание контента, перевод текстов, услуги персонального ассистента, суммаризация, поиск [1]. LLM изначально обучались выполнять подобного рода задачи. Несмотря на ограниченность разнообразных задач в обучающих выборках, в процессе тренировок нейросети научились выполнять те запросы пользователей, которые они точно видят впервые [2, 3].

2. LegalTech. Использование нейросетей в юриспруденции может быть крайне полезным, ввиду того, что значимая часть работы юристов так или иначе связана с анализом и написанием текстов, что, в свою очередь, и является основной задачей, которую решают большие языковые модели. Такие задачи, как составление договоров, анализ юридических документов, оценка рисков, хоть пока и не могут быть решены полностью при помощи LLM, но при правильном и контролируемом использовании могут сократить время работы юристов в разы [4].

На данный момент подавляющее большинство решений, представленных на рынке LegalTech [5, 6] – это специально обученные под

Смирнов Даниил Олегович – магистрант, Санкт-Петербургский государственный университет; e-mail: st077086@student.spbu.ru, тел.: +7(999)123-41-39

Блеканов Иван Станиславович – доцент, Санкт-Петербургский государственный университет; e-mail: i.blekanov@spbu.ru, тел.: +7(921)339-53-43

Исследование проведено в рамках проекта «Накопительное общественное мнение и кумулятивная делиберация: теоретизация, методы анализа и оценка качества современного делиберативного процесса», ID 128786104, 2025

юридический домен LLM: при составлении тренировочного датасета их весомая часть состояла из решаемых в будущем юридических задач. Сложность заключается в том, что обучить подобную нейросеть с нуля могут лишь компании, обладающие огромным количеством вычислительных мощностей [7]. В данной работе акцент сделан на методы, которые позволяют генерализированную LLM, имеющую недостаточные метрики качества в выбранной области, адаптировать для улучшения качества генераций без использования для этого большого количества вычислительных ресурсов.

3. Методы адаптации. Для начала был рассмотрен стандартный подход к генерации ответов LLM.

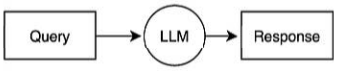


Рис. 1. Стандартный pipeline

На вход языковой модели подается запрос пользователя, на который модель генерирует ответ до тех пор, пока не закончится контекст, либо пока не будет встречен специальный токен, сигнализирующий об окончании предложения (рис. 1).

3.1. Prompt Engineering. Это метод адаптации LLM, который не требует дополнительных обучающих данных. Он представляет собой ручной подбор оптимального запроса к нейросети с целью улучшения понимания моделью постановки задачи (рис. 2). Существует большое количество разнообразных техник prompt engineering: Few-Shot Prompting, Chain-of-Thought, Self-Consistency и т. д. [8].

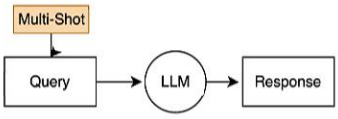


Рис. 2. Prompt Engineering pipeline

В данном исследовании рассматривался метод Few-Shot. Его главная идея состоит в том, чтобы дать модели несколько примеров того, что от нее требуется. Для примера рассмотрим задачу анализа тональности текста: определение того, является ли фраза негативной, позитивной или нейтральной. Тогда запрос к модели в формате Few-Shot может выглядеть следующим образом:

```
This is awesome! // Positive
This is bad! // Negative
```

```
Wow that movie was rad! // Positive
```

3.2. Low-Rank Adaptation (LoRA). Это метод адаптации нейросети, который позволяет существенно сократить вычислительные затраты при дообучении языковой модели [9]. Вместо изменения всех весов сети данный метод предлагает добавлять небольшое количество параметров (доли процентов от общего числа весов) в виде низкоранговых матриц, которые модифицируются в процессе тренировки, в то время как исходные параметры модели остаются замороженными (рис. 3).

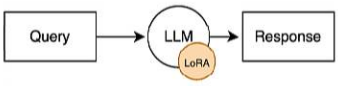


Рис. 3. LoRA pipeline

Применительно к LegalTech данный метод может быть особенно полезен, поскольку позволяет адаптировать LLM к специализированной терминологии, стилю изложения и логике юридических документов без необходимости полного дообучения. Кроме того, благодаря модульности LoRA можно эффективно переключаться между различными версиями адаптированных параметров, что позволяет использовать одну и ту же базовую модель для множества юридических юрисдикций или типов контрактов.

3.3. Retrieval-Augmented Generation (RAG). Это метод адаптации моделей, сочетающий в себе генеративные возможности LLM и извлечение релевантной информации из внешних источников [10]. Его идея состоит в том, что перед началом генерации ответа в контекст модели подкладывают информацию из имеющихся источников, которая могла бы быть полезна для решения задачи (рис. 4).

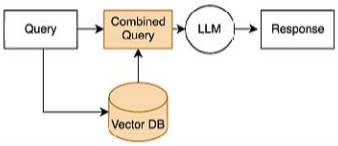


Рис. 4. RAG pipeline

В домене LegalTech данная техника будет крайне полезна, к примеру, при правовой экспертизе. В процессе обработки документа в контекст нейросети будут добавляться требуемые статьи из уголовного или конституционного кодекса, вследствие чего модель будет меньше галлюцинировать и выдавать более корректные ответы.

4. Эксперимент. Одной из важных задач в LegalTech является проверка логической согласованности юридических документов. Чтобы проверить качество модели при решении данной задачи, в данном исследовании использовался бенчмарк Contract-NLI [11].

Ответом модели считалось первое вхождение метки (contradiction, entailment, neutral) в текст. Так как все метки классов равнозначны, но при этом присутствует дисбаланс классов, для оценки качества моделей использовалась метрика Macro F1-score.

В качестве языковых моделей использованы три open-source решения:

1. LLaMa-3.2-3b-instruct от компании MetaAI [12].
2. Gemma-2-2b-it от компании Google [13].
3. Qwen2.5-3B-Instruct от компании Alibaba [14].

Для классификации использовался следующий промпт:

```
You are an expert in legal contract analysis.
Your task is to determine the relationship between a premise
(a clause from a contract) and a hypothesis
(a statement about that clause).
For each pair, classify the relationship as one of:

- **Entailment**: The hypothesis necessarily
    follows from the premise.
- **Contradiction**: The hypothesis contradicts
    the premise.
- **Neutral**: The premise neither entails nor contradicts
    the hypothesis.

**Premise**: [PREMISE]
**Hypothesis**: [HYPOTHESIS]
**Classification**:
```

4.1. Prompt Engineering. В данном методе к вышеописанному промпту были добавлены следующие примеры решения задачи.

```
### Example 1:
**Premise**: "The Consultant shall maintain the
```

```
confidentiality of all proprietary information disclosed by
the Company for a period of five years after the
termination of this Agreement."
**Hypothesis**: "The Consultant must keep the Company's
proprietary information confidential for
5 years after the agreement ends."
**Classification**: Entailment

### Example 2:

**Premise**: "Either party may terminate this Agreement with
thirty (30) days written notice." **Hypothesis**: "Only the
Company can terminate the Agreement."
**Classification**: Contradiction

### Example 3:

**Premise**: "Payments shall be made within fifteen (15)
business days of receipt of invoice." **Hypothesis**:
"Payments will be processed through an electronic
transfer system."
**Classification**: Neutral
```

4.2. LoRA. Для каждой модели был обучен свой LoRA адаптер, гиперпараметры к ним указаны в таблице 1. В качестве обучающей выборки использовались два датасета:

- SNLI [15] – чтобы модель лучше понимала подтверждения, противоречия и отсутствие связи между утверждениями;
- CUAD [16] – чтобы модель лучше разбиралась в домене LegalTech.

Таблица 1. Гиперпараметры LoRA адаптера

№ Гиперпараметр	Значение
learning_rate	10^{-4}
batch_size	16
epochs	3
lora_rank	8
lora_alpha	16
lora_dropout	0,1

4.3. RAG. В качестве данных для RAG системы также использовались датасеты SNLI и CUAD. Векторные представления семплов строились при помощи модели all-mpnet-base-v2. Для оптимального поиска ближайших соседей использовалась база данных PostgreSQL, а на столбец с векторным представлением был поставлен индекс HNVS (Hierarchical Navigable Small World), который позволяет искать ближайших соседей со сложностью $O(\log n)$.

Промпт для Retrieval-Augmented Generation:

```
Use the provided external data to generate your response.
Prioritize accuracy and relevance by incorporating
information from the retrieved documents. If the
external data is insufficient, clarify uncertainty
rather than guessing.

[EXTERNAL_DATA]

You are an expert in legal contract analysis. ...
```

4.4. Результаты эксперимента. В таблице 2 представлены метрики базовых и адаптированных моделей на датасете Contract-NLI. Наилучшие метрики двух из трех моделей показало использование RAG-систем, однако для Qwen2.5-3b-Instruct наилучшим методом оказалась LoRA.

Таблица 2. Метрики моделей на датасете Contract-NLI

№ Model Name	Base	Prompt Engineering	LoRA	RAG
LLaMa-3.2-3b-instruct	0,43	0,43	0,50	**0,51**
Gemma-2-2b-it	0,58	0,60	0,59	**0,63**
Qwen2.5-3B-Instruct	0,50	0,49	**0,54**	0,51

5. Заключение. Результаты исследования показали, что представленные методы адаптации LLM, в большинстве случаев, оказывают положительное влияние на точность ответов моделей.

Для двух моделей наиболее значимого прироста получилось добиться при использовании RAG-систем. Тем не менее, LoRA также является эффективным способом адаптации, практически не уступающим RAG по метрикам. Prompt Engineering, в свою очередь, хоть и не дает столь значимого прироста метрик, но остается наименее затратным и полезным инструментом для улучшения ответов модели.

Литература

1. Naveed H. A comprehensive overview of large language models // arXiv.org. 2023. Дата обновления: 12.07.2023. [Электронный ресурс]: URL:https://arxiv.org/abs/2307.06435 (дата обращения: 17.03.2025).

2. Wei J. Emergent abilities of large language models // arXiv.org. 2022. Дата обновления: 15.06.2022. [Электронный ресурс]: URL:https://arxiv.org/abs/2206.07682 (дата обращения: 17.03.2025).

3. B. Brown T. Language models are few-shot learners // arXiv.org. 2020. Дата обновления: 28.05.2020. [Электронный ресурс]: URL:https://arxiv.org/abs/2005.14165 (дата обращения: 17.03.2025).

4. Savelka J. Explaining legal concepts with augmented large language models (GPT-4) // arXiv.org. 2023. Дата обновления: 15.06.2023. [Электронный ресурс]: URL:https://arxiv.org/abs/2306.09525 (дата обращения: 17.03.2025).

5. GigaLegal – виртуальный юрист на базе GigaChat // GigaLegal. 2025. Дата обновления: 10.01.2025. [Электронный ресурс]: URL:https://developers.sber.ru/help/gigachat-api/legal-consultation (дата обращения: 17.03.25).

6. Кузнецов М. М. Разработка сервиса юридической помощи на основе методов машинного обучения // Процессы управления и устойчивость. 2022. Т. 9. № 1. С. 248–253.

7. Kaplan J. Scaling laws for neural language models // arXiv.org. 2020. Дата обновления: 23.01.2020. [Электронный ресурс]: URL:https://arxiv.org/abs/2001.08361 (дата обращения: 17.03.2025).

8. Sahoo P. A systematic survey of prompt engineering in large language models: techniques and applications // arXiv.org. 2024. Дата обновления: 05.02.2024. [Электронный ресурс]: URL:https://arxiv.org/abs/2402.07927 (дата обращения: 17.03.2025).

9. J. Hu E. LoRA: Low-rank adaptation of large language models // arXiv.org. 2021. Дата обновления: 17.06.2021. [Электронный ресурс]: URL:https://arxiv.org/abs/2106.09685 (дата обращения: 17.03.2025).

10. Lewis P. Retrieval-augmented generation for knowledge-intensive NLP tasks // arXiv.org. 2020. Дата обновления: 22.05.2020. [Электронный ресурс]: URL:https://arxiv.org/abs/2005.11401 (дата обращения: 17.03.2025).

11. Koreeda Y. ContractNLI: a dataset for document-level natural language inference for contracts // arXiv.org. 2021. Дата обновления: 05.10.2021. [Электронный ресурс]: URL:https://arxiv.org/abs/2110.01799 (дата обращения: 17.03.2025).

12. Touvron H. LLaMA: open and efficient foundation language models // arXiv.org. 2023. Дата обновления: 27.02.2023. [Электронный ресурс]: URL:https://arxiv.org/abs/2302.13971 (дата обращения: 17.03.2025).

13. Gemma Team. Gemma: open models based on gemini research and technology // arXiv.org. 2024. Дата обновления: 13.03.2024. [Электронный ресурс]: URL:https://arxiv.org/abs/2403.08295 (дата обращения: 17.03.2025).

14. Bai J. Qwen technical report // arXiv.org. 2023. Дата обновления: 28.09.2023. [Электронный ресурс]: URL:https://arxiv.org/abs/2309.16609 (дата обращения: 17.03.2025).

15. R. Bowman S. A large annotated corpus for learning natural language inference // arXiv.org. 2015. Дата обновления: 21.08.2015. [Электронный ресурс]: URL:https://arxiv.org/abs/1508.05326 (дата обращения: 17.03.2025).

16. Hendrycks D. CUAD: An expert-annotated NLP dataset for legal contract review // arXiv.org. 2021. Дата обновления: 10.03.2021. [Электронный ресурс]: URL:https://arxiv.org/abs/2103.06268 (дата обращения: 17.03.2025).

УДК 519.17+004.032.26

Снапков М. Ю., Шмаков А. А.

Применение данных социальных сетей в задачах анализа финансовых инструментов

Рекомендовано к публикации доцентом Блекановым И. С.

Введение. Социальные сети оказывают значительное влияние как на отдельных людей и организации, так и на общество в целом. Постоянное увеличение числа социальных сетей, а также количества пользователей, каждый из которых оставляет свой след в виде сообщений или другого контента, приводит к стремительному росту объёмов данных. Это создаёт спрос на улучшение и разработку новых методов анализа социальных сетей (SNA, Social Network Analysis).

В связи с активным взаимодействием пользователей на множестве платформ возникают ситуации, когда такое взаимодействие перестаёт быть простым общением и может нести угрозы безопасности, а также влиять на различные процессы, включая политические, экономические и другие. Исследование этого влияния может открыть новые перспективы для анализа, например, в контексте изучения фондовых рынков.

Модель описания дискуссии. В статьях [1, 2] подробно описано моделирование дискуссий социальных сетей. Дискуссия D_t в рамках выбранной социальной сети (Reddit) – это множество пользователей U_t^A, которые инициируют дискуссию, генерируя сообщения $M_t^{U_t^A}$, в момент времени t:

$$D_t = \{U_t^A, M_t^{U_t^A}\}, \quad t \in [0, T].$$

Множество сообщений $M_t^{U_t^A}$ является не просто текстами, а тесно связано с пользователями, участвующими в обсуждении $U_{t'}$:

$$U_{t'} = \{U_t^{AC}, U_{t'}^C\},$$

Снапков Максим Юрьевич – магистрант, Санкт-Петербургский государственный университет; e-mail: st087381@student.spbu.ru, тел.: +7(981)447-12-73

Шмаков Александр Андреевич – магистрант, Санкт-Петербургский государственный университет; e-mail: st085954@student.spbu.ru, тел.: +7(920)929-89-74

где U_t^{AC} – это подмножество таких пользователей U_t^A, которые не только инициируют дискуссию, но ещё и оставляют комментарии; $U_{t'}^C$ – множество только комментирующих пользователей. Комментарии участвующих в дискуссии пользователей обозначаются $C_{t'}^{U_{t'}}$, а их реакции на сообщения – $L_{t'}^{U_{t'}}$ (отметка «нравится» или «не нравится»). Тогда

$$M_t^{U_t^A} = \{U_{t'}, C_{t'}^{U_{t'}}, L_{t'}^{U_{t'}}\}, \quad t' \in [t, T].$$

Анализ социального графа. Для работы с сообществами в социальных сетях используются методы, которые основаны на анализе социального графа. Социальный граф – это граф, узлами которого являются пользователи, а рёбра графа отображают взаимодействия между ними. С учетом особенностей Reddit, структура графа представлена следующим образом: ребро с весом n и направлением от узла «А» к узлу «Б» говорит о том, что пользователь А оставил n комментариев под публикациями пользователя Б.

Для анализа таких графов можно использовать различные меры центральности, которые помогают определить важность узлов в сети. Одной из таких мер является Betweenness – центральность по посредничеству. Она показывает, насколько часто узел лежит на кратчайших путях в графе. В документации [3] представлено описание встроенной функции betweenness_centrality библиотеки $NetworkX$ для Python. Формула для вычисления центральности по посредничеству выглядит следующим образом:

$$c_B(v) = \sum_{s,t \in V} \frac{\sigma(s,t|v)}{\sigma(s,t)},$$

где V – множество узлов, $\sigma(s,t)$ – количество кратчайших путей (s,t), а $\sigma(s,t|v)$ – количество путей, проходящих через узел v. Если $s = t$, то $\sigma(s,t) = 1$, и если $v \in s,t$, то $\sigma(s,t|v) = 0$.

Другой важной метрикой является PageRank [4] – числовая величина, характеризующая «важность» страницы. Чем больше других страниц ссылается на рассматриваемую, тем она «важнее». Кроме того, «вес» страницы A определяется весом ссылки, передаваемой страницей B. Таким образом, PageRank – это метод вычисления веса узла путём подсчёта важности ссылок на него. Формула для вычис-

ления PageRank выглядит следующим образом:

$$R'(u) = (1-d) + d \sum_{v \in B_u} \frac{R'(v)}{N_v},$$

где $R'(u)$ – PageRank рассматриваемой страницы u, d – коэффициент затухания (означает вероятность того, что пользователь, зашедший на страницу, перейдет по одной из ссылок, содержащейся на этой странице, а не прекратит путешествие по сети), в классической формуле обычно $d = 0{,}85$; B_u – множество страниц, ссылающихся на рассматриваемую, а N_v – общее число ссылок на странице v.

Пользователей, имеющих высокие значения по обеим мерам центральности, можно описать следующим образом: это пользователи, которые часто связывают различные подсообщества (мера Betweenness отражает, насколько часто узел графа лежит на кратчайшем пути между двумя другими узлами), а также с ними чаще взаимодействуют другие «важные» пользователи (ранжирование с помощью PageRank). В дальнейшем такие пользователи будут именоваться «инфлюенсерами».

Методы анализа тональности и обработки пользовательского контента. Методы машинного обучения из области обработки естественного языка позволяют решать различные задачи, от классификации до генерации текста, включая оценку тональности. В настоящее время для решения этих задач активно используются предобученные модели, такие как BERT. В статьях [5, 6] авторы использовали модель FinBERT [7] (Financial Bidirectional Encoder Representations from Transformers) – модель NLP (Natural Language Processing) для анализа тональности текстов, связанных с финансами. Модель основана на языковой модели BERT и дообучена на большом наборе финансовых данных, что позволяет эффективно классифицировать тональность финансовых текстов. Основной целью авторов статьи [5] был анализ данных социальной сети Twitter, связанных с биткойнами, а в статье [6] исследуются заголовки финансовых новостей. На основе описаний исследований можно сделать вывод, что модель FinBERT применима для определения тональности публикаций пользователей социальных сетей на финансовые темы, что является её ключевым преимуществом при выборе модели для анализа тональности. Если на вход предобученной модели BERT, дообученной на данных с финансовым контекстом, подать пользователь-

ский комментарий c_i, на выходе будет получена оценка тональности s_{c_i} для данного комментария. Здесь $c_i \in C_t$, где C_t – множество всех комментариев за определённый временной интервал t, написанных в социальной сети в рамках публичной дискуссии. Значение оценки $s_{c_i} \in [-1, 1]$ можно интерпретировать как вероятность принадлежности комментария к одному из заданных классов: негативный (-1), положительный (1) или, в некоторых случаях, нейтральный (0).

Установив определённые пороговые значения, можно присвоить каждому комментарию метку класса $\overline{s}_{c_i}$, где $\overline{s}_{c_i} \in \{-1, 1\}$.

Используя оценки тональности s_{c_i} и $\overline{s}_{c_i}$, можно ввести новые показатели [8]:

$$T_t = \sum_i s_{c_i},$$

$$SI = \ln \frac{1 + N_+}{1 + N_-},$$

$$SDI = \left| 1 - \left| \frac{N_+ - N_-}{N_+ + N_-} \right| \right|,$$

где N_+ – количество позитивных комментариев, N_- – количество негативных комментариев. Нормирование показателя T_t позволит оценить общий тренд тональности за временной интервал t.

Эксперименты и результаты. На основе данных, полученных в результате анализа социальных сетей были проведены эксперименты, в которых построена модель, прогнозирующая движение акции, а также обнаружена взаимосвязь влияния инфлюенсеров и общего тренда стоимости финансового инструмента.

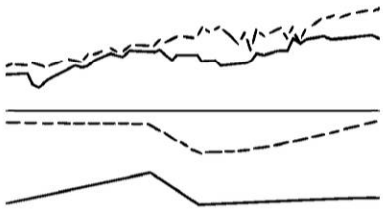


Рис. 1. Прогнозирование динамики цены

В эксперименте модель Transformer обучалась предсказывать направление движения стоимости акции. На рис. 1 представлен резуль-

тат тестирования модели на тестовой (верхний график) и валидационной (нижний график) выборках данных, где линией изображён график стоимости акции SBER, а пунктиром – предсказанное моделью значение.

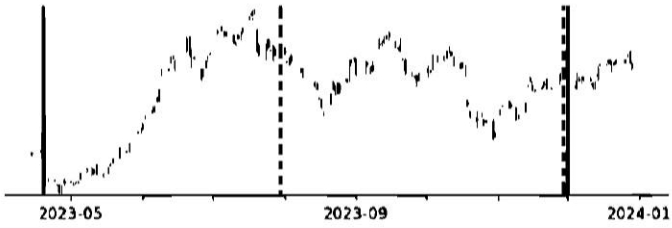


Рис. 2. Динамика тональности дискуссии

На рис. 2 изображён свечной график, демонстрирующий зависимость между активностью инфлюенсеров и динамикой движения акции, которую удалось отследить в эксперименте, используя данные об активности пользователей в социальной сети Reddit относительно акций TSLA. Вертикальные линии и пунктиры означают позитивную и негативную активность инфлюенсера соответственно. Можно заметить, что использование в совокупности таких подходов может позволить улучшить качество прогноза модели. С результатами экспериментов можно ознакомиться в [9, 10].

Заключение. Таким образом, представленные метрики и показатели были получены на основе данных о пользовательской активности в социальных сетях и могут внести значительный вклад в повышение качества анализа фондовых рынков и других аналогичных систем. Использование данных социальных сетей в области финансовых инструментов может быть полезным для моделирования рыночной динамики, включая прогнозирование аномалий, поскольку позволяет количественно интерпретировать такие сложные факторы, как общественное настроение и вклад отдельных пользователей в его формирование.

Литература

1. Blekanov I., Bodrunova S. S., Akhmetov A. Detection of hidden communities in twitter discussions of varying volumes // Future Internet. 2021. Vol. 13. No 11. P. 295.

2. Проноза А. А., Виткова Л. А., Чечулин А. А., Котенко И. В., Сахаров Д. В. Методика выявления каналов распространения информации в социальных сетях // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2018. Т. 14. № 4. С. 362–377.

3. Betweenness_centrality – NetworkX 3.1 documentation // NetworkX Network Analysis in Python. [Электронный ресурс]: URL:https://networkx.org/documentation/stable/reference/algorithms/generated/networkx.algorithms.centrality.betweenness_centrality.html (дата обращения: 15.05.2023).

4. Page L., Brin S. et al. The PageRank citation ranking: Bringing order to the web. // Stanford infolab. 1999.

5. Srivastava N., Gowda O. et al. Bitcoin evolution analytics: Twitter sentiments to predict price change as bearish or bullish // 2022 IEEE International Conference on Big Data (Big Data). 2022. P. 3547–3556.

6. Sidogi T., Mbuvha R., Marwala T. Stock price prediction using sentiment analysis // 2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC). 2021. P. 46–51.

7. Araci D. Finbert: Financial sentiment analysis with pre-trained language models // arXiv preprint arXiv:1908.10063. 2019.

8. Wang Y., Wang Y. Using social media mining technology to assist in price prediction of stock market // 2016 IEEE International Conference on Big Data Analysis (ICBDA). 2016. P. 1–4.

9. GitHub [Электронный ресурс]: URL:https://github.com/maxsnapkov/diploma (дата обращения: 19.03.2025).

10. GitHub [Электронный ресурс]: URL:https://github.com/AlexSHDEV/stock_prediction_VKR (дата обращения: 19.03.2025).

УДК 004.75

Соловьев Д. Н.

Система домашней автоматизации с использованием децентрализованного подхода в хранении и обработке данных

Рекомендовано к публикации доцентом Корховым В. В.

1. Введение. С увеличением количества устройств интернет вещей (ИВ) в системах домашней автоматизации (СДА) остро встает вопрос топологий сетей, в которых находятся эти устройства, а также способах хранения и передачи данных между устройствами. Многим системам, использующим централизованный подход в хранении и обработке информации, свойственна существенная потеря функциональности при выходе из строя центрального узла. В ходе исследования были изучены принципы, методологии и архитектуры, применимые к программно-аппаратным комплексам для построения СДА и позволяющие повысить уровень безопасности данных, отказоустойчивость и увеличить скорость передачи информации в рамках системы за счет использования распределенного подхода к хранению и обработке информации, разработан прототип распределенной СДА. Результаты проведенного исследования и предложенная в нем архитектурная модель могут быть использованы не только в системах умных домов, но и в умных городах и на предприятиях.

2. Анализ существующих решений. В данный момент большинство производителей ИВ используют централизованный подход для построения СДА [1]. В качестве центра умного дома (ЦУД) выступает устройство (хаб), объединяющее конечные устройства в единую сеть. Сценарии взаимодействия устройств могут быть локальными (сценарии хранятся в ЦУД) или облачными. Вариант с локальным расположением сценариев более предпочтительный, так как повышается отзывчивость системы и ее отказоустойчивость. Отдельную сложность представляет объединение устройств из разных экосистем: различные комбинации связей «устройство», «ЦУД», «облако». Большинство имеющихся на данный момент решений опира-

Соловьев Дмитрий Николаевич – студент, Санкт-Петербургский государственный университет; e-mail: st094239@student.spbu.ru, тел.: +7(906)484-77-55

ются на топологию «дерево», где в качестве корня рассматривается ЦУД или облако, что проиллюстрированно на рис. 1. При этом структура может подразумевать наличие единственных ветвей из корня, например «облако – ЦУД – устройства» [2]. Тенденции 2023–2024 г. по переносу обработки событий в ЦУД, который ранее использовался лишь для маршрутизации сообщений от устройств к облаку, позволили увеличить надежность и уменьшить задержки в системе, но не исключили имеющиеся проблемы полностью.

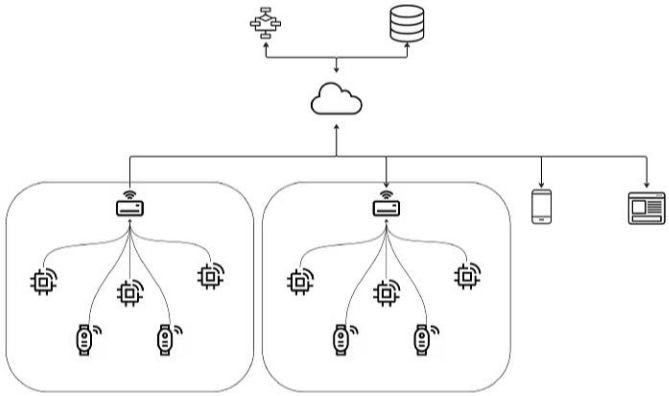


Рис. 1. Пример СДА с облачной обработкой событий

Стоит различать аппаратные ограничения, не позволяющие изменить топологию системы, связанные с особенностью работы протокола, например, топологию Bluetooth, позволяющую организовывать только связи типа точка – точка или точка – множество устройств, и архитектуру СДА, подразумевающую централизованное выполнение сценария в ЦУД либо облаке.

Также ограничения по топологии присутствуют и в стандартах сред беспроводных сетей, таких как IEEE 802.15.4, IEEE 802.11 [3] и т. д., в частности, необходимость использования координатора сети, выход из строя которого полностью блокирует передачу сообщений в сети.

3. Постановка задачи. После анализа существующих решений и выявления недостатков СДА, присутствующих на рынке, были сформулированы начальные условия (требования).

1. Использование одноуровневой сети для связи устройств в рамках единой зоны.
2. Взаимодействие устройств осуществляется напрямую, без посредников в виде ЦУД.
3. Взаимодействие устройств из разных зон обеспечивается благодаря шлюзам-ретрансляторам, обединенными в единую сеть.

Поставлена задача: спроектировать и разбработать прототип СДА, определить варианты реализаций и, проведя ряд экспериментов, оценить преимущества и недостатки представленного подхода.

4. Вариант реализации и прототипирование. Централизованные СДА, имеющие ЦУД, обладая несомненными преимуществами, такими как простота настройки и реализации, также имеют ряд недостатков: низкая скорость реакции, невысокая надежность и безопасность, сильно зависящая от ЦУД и облачной инфраструктуры. Для интеграции устройств, находящихся в разных сетях, используется централизованная координация через единое облако. На рис. 2 представлена архитектура предлагаемой гибридной системы, подразумевающей прямое взаимодействие устройств в СДА.

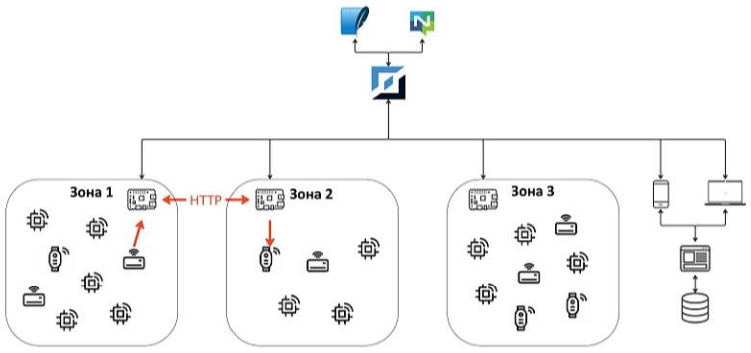


Рис. 2. СДА с распределенной структурой

Для взаимодействия физически разнесенных устройств используется кластер K0s с несколькими координаторами сети. Данный кластер в каждой зоне содержит по одному узлу, объединяющему зоны в единую сеть, распределяющему по устройствам обновленную

конфигурацию системы, а также позволяющему управлять СДА через графический интерфейс.

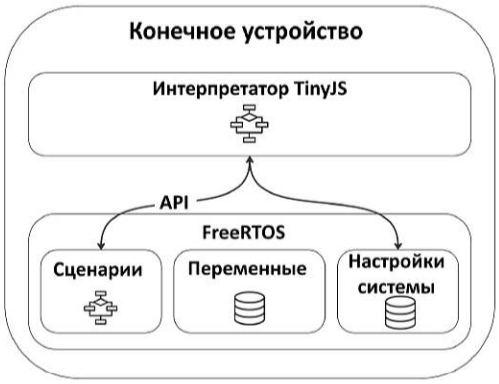


Рис. 3. Архитектура конечного устройства

Програмная часть каждого конечного устройства представляет из себя двухуровневую систему, как показано на рис. 3: низкоуровневую прошивку, написанную с использованем FreeRTOS и предоставляющую API для высокоуровневой части для управления, и высокоуровневую прошивку, написанную на языке JavaScript, запускающуюся посредством интерпретатора на конечном устройстве. JS-часть может быть легко изменена через настройку системы. В рамках одной зоны предложенная система позволяет отказаться от ЦУД, обеспечивая обработку устройствами-актуаторами событий, поступающих напрямую от устройств-сенсоров. Данное решение кроме повышения надежности системы также уменьшает задержку передачи информации, повышает отказоустойчивость системы и ее безопасность [4, 5].

5. Тестирование и полученные оценки. В связи со сложностью разработки прототипа на основе беспроводной сети Zigbee, были использованы радиомодули SI4432, что делает прямое тестирование СДА, основанных на разных беспроводных протоколах, непоказательным, поскольку различие в протокольных стеках и принципах работы модулей исключает возможность корректного сопоставления характеристик систем.

Для корректности тестирования и сравнимости подходов, ввиду

отличия протоколов передачи данных, использования разных микроконтроллеров в существующих системах и иных отличий, исходная система была переконфигурирована для работы как с ЦУД, так и без ЦУД. Таким образом, проведя замеры задержек от возникновения события до реакции, можно оценить численно преимущество предложенной архитектуры. В таблице приведены результаты замеров на каждом этапе: время передачи пакета сообщения, время обработки события для каждой системы, что позволило использовать эти данные без привязки к конкретной реализации. Были проведены замеры времени задержек выполнения программы на микроконтроллере STM32F411 (25 МГц), выступавшем в качестве конечного устройства, и одноплатным компьютером Raspberry pi 3b+. Тестирование одноплатного компьютера проводилось на системе Raspberry Pi OS 19.11, базирующейся на Debian 12. Использовался GCC 15 с опциями: `-mcpu=native, -O2`. При тестировании микроконтроллеров использовалась STM32duino FreeRTOS v10.3.2 с библиотекой RadioLib v7.1.2. Радиомодули, представленные SI4432, использовались со стандартными параметрами.

Таблица. Среднее время выполнения операций

№ п/п	Тип операции	Время
1	Передача в рамках одной зоны	197мс
2	Выполнения 1 примитива интерпретатора на микроконтроллере	18 мс
3	Выполнения 1 примитива интерпретатора на одноплатном компьютере	0,24 мс
4	Общая задержка системы при использовании ЦУД	407 мс
5	Общая задержка системы без использования ЦУД	231 мс
6	Задержка СУД «Умный дом с Алисой» от Яндекс	750 мс

Использовалось готовое решение на базе протокола Zigbee с ЦУД в виде YNDX-00510 [6] и лампой YNDX-00558. Лучшим способом сокращения задержек в СУД является уменьшение задержек в передаче информации. Предложенный выше подход позволяет почти в вдвое сократить задержку (в рамках тестирования спроектированной СУД) и более чем втрое по сравнению с представленными на рынке решениями.

6. Заключение. Предложен подход к построению распределенной системы домашней автоматизации, проведено сравнение с решением, представленным на рынке. В дальнейшем предполагается исследовать возможные уязвимости данной системы, построить и протестировать прототип распределенной системы с использованием шлюзов-ретрансляторов, объединенных в единую сеть посредством VPN.

Литература

1. Шерчков А. В. Анализ существующих технических решений «Умный дом» // Молодежный вестник Уфимского государственного авиационного технического университета. 2020. № 1. С. 153–155.

2. Тукмачева Ю. А. Обзор и анализ автоматизированных систем «Умный дом», представленных на российском сегменте рынка // Вестник науки. 2022. Т. 4. № 10. С. 139–144.

3. Колыбельников А. И. Обзор технологий беспроводных сетей // ТРУДЫ МФТИ. 2012. Т. 4. № 2. С. 3–29.

4. Некрасов П. В., Жариков А. М., Козин Д. А. Обеспечение безопасности беспроводных каналов связи киберфизических систем типа «Умный дом» // Безопасность информационных технологий. 2024. Т. 31. № 1. С. 54–62.

5. Vardakis G., Hatzivasilis G., Koutsaki E., Papadakis N. Review of smart-home security using the internet of things // Electronics. 2024. No 13. P. 33–43.

6. Центр умного дома «Яндекс Хаб» модель YNDX-00510. Инструкция по эксплуатации [Электронный ресурс]: `URL:https://yastatic.net/s3/doc-binary/src/support/smart-home/hub/hub-manual.pdf` (дата обращения: 17.03.2025).

УДК 004.852

Чайкин Г. А.

Создание синтетических данных пользовательской активности на основе вопросно-ответных текстовых данных

Рекомендовано к публикации доцентом Блекановым И. С.

1. Введение. Данные являются важной частью создания алгоритма машинного обучения, но не всегда их возможно получить. Кроме того, данных может быть недостаточно для обучения модели, или возникает вопрос их обработки, заполнения пустых данных или их анонимизации. Если данных нет или их мало, то одним из вариантов решения проблемы является использование синтетических данных. В рамках задачи отслеживания знаний задача создания синтетической выборки может усложниться за счет того, что модель на вход получает данные пользовательской активности, которые имеют последовательную структуру. Универсальным методом генерации может выступить метод на основе большой языковой модели, у которой достаточно обобщающей способности для решения задач, которые не фигурировали во время её обучения [1]. Надстройка под конкретную задачу может выполняться путем уточнения начальной инструкции в промпте, а также путем добавления различных примеров к инструкции.

Данная работа посвящена процессу создания синтетической выборки, а также тому, как может обучение с подобными данными повлиять на обобщающей способности модели отслеживания знаний на реальных данных.

2. Задача отслеживания знаний. Задача отслеживания знаний является задачей обучения с учителем, где в качестве данных используется пользовательская активность учащихся. Её можно представить в виде последовательности ответов на задания. Эти последовательности и используются в качестве признаков. Меткой для предсказания является ответ на следующий вопрос.

Чайкин Георгий Александрович – аспирант, Санкт-Петербургский государственный университет; e-mail: st061320@student.spbu.ru, тел.: +7(981)755-59-34

Пусть задано множество последовательностей $X = \{(x_t)_{t=1}^{T_i} : T_i \in N, i \in N\}$, где $x_t = (q_t, a_t)$ – это взаимодействие учащегося с платформой, q_t – это вопрос для учащегося на шаге t, а a_t – это ответ учащегося на вопрос q_t. Также задано множество допустимых ответов Y, и существует целевая функция $y^* : X \to Y$, значения которой $y_i = y^*\left((x_t)_{t=1}^{T_i}\right)$ известны не только на конечном подмножестве объектов $X_l = \left\{(x_t)_{t=1}^{T_i}\right\}, i = \overline{1,l}$, но и на всем множестве X. В задаче отслеживания знаний $y_i \in Y$ – это некоторый аспект следующего взаимодействия x_{t+1}. Конкретно в данной задаче $y = [0, 1]$, где $y \in Y$ – это условная вероятность правильного ответа на следующем вопросе $p(a_{t+1} = 1|q_{t+1}, X)$.

Полученные результаты можно использовать для разработки интеллектуальных систем обучения, для рекомендации учебных материалов [2] или для определения начальных данных для задачи распределения заданий, определенной в [3].

3. Исходные данные для генерации выборки. В качестве первоначальных данных для генерации выборки был выбран датасет MATH [4]. Эта выборка, предназначенная для решения математических задач с использованием методов машинного обучения. Датасет охватывает широкий спектр математических тем и уровней сложности, начиная от простых арифметических задач и заканчивая более сложными заданиями из областей алгебры, геометрии, теории чисел и комбинаторики. В данной работе в качестве исходной выборки будут использоваться алгебраические задачи.

4. Процесс создания синтетических данных. Процесс создания синтетических данных включает в себя несколько этапов.

На первом этапе происходит преобразование данных. Изначально они представляют собой список .json документов. Код последовательно извлекает вопросы и ответы и отправляет их для дальнейшей обработки.

Второый этап состоит из генерации текста на основе вопросно-ответных пар. Пары передаются последовательно, тем самым накапливая историю пользовательской активности. В качестве пользователя выступает большая языковая модель. После того, как она сгенерирует ответ, результаты (а также вопрос и правильный ответ) отправляются на проверку другой модели, которая генерирует бинарную метку, обозначающую, правильно ли пользователь отве-

тил на вопрос или нет. Процесс подготовлен на базе работы [5]. В качестве моделей используются gpt-4o-mini и gpt-4o для ответов на вопросы и проверки ответов соответственно.

Последний этап состоит из обработки и сохранения синтетических данных. Во время каждой итерации генерации ответа и бинарной метки они сохраняются в виде .csv файла.

5. Численные эксперименты. На основе подготовленного кода реализован описанный выше метод создания синтетических данных, и были собраны синтетические данные пользовательской активности. В итоге получилось 10 последовательностей по 50 строк с бинарными метками, которые обозначают, правильно ли модель ответила на вопрос. Дополнительно были собраны 5 последовательностей по 50 строк, для каждой из которых в инструкции были предоставлены примеры с решением задач.

Чтобы проверить результат генерации, было решено обучить специальную параметрическую модель, которая в рамках задачи отслеживания знаний может использовать сгенерированные бинарные метки, под названием Bayesian Knowledge Tracing (BKT) [6].

Для экспериментов была применена библиотека pyBKT [7] на языке Python. Она содержит необходимые модификации BKT для текущей работы.

В качестве тестовых данных взята отдельная выборка [8], которая содержит данные пользовательской активности решения математических задач. Датасет был предварительно отфильтрован, чтобы в нем оказались только данные по алгебре, а также разделен на тренировочную (чтобы обучить модель и сравнить с моделью на синтетических данных) и тестовую подвыборки в соотношении 80% и 20%. В итоге получилось 207860 и 54561 строк соответственно.

В качестве тестовых метрик выступают точность (Accuracy) и площадь под кривой (AUC). Результаты экспериментов представлены в таблице.

Таблица. Метрики Accuracy и AUC на тестовой выборке

	Accuracy	**AUC**
BKT (Assistments)	0,62	0,60
BKT (Synthetic data)	0,69	0,61
BKT (Synthetic data, Few-shot)	0,62	0,41
BKT (All synthetic data)	0,62	0,61

В первой строке представлены метрики для модели, обученной на тренировочной выборке из [8]. Во второй строке – для модели, обученной на синтетических данных. По метрикам вторая модель превосходит первую даже с учетом того, что синтетических обучающих данных в разы меньше (500 строк против 207860).

В третьей строке представлены результаты модели, которая также обучалась на синтетических данных, но в данном случае они собирались на основе инструкции с дополнительными примерами вопросов и ответов (в инструкции были представлены примеры с решениями 5 алгебраических задач). По метрикам модель ведет себя хуже остальных. Если же объединить синтетические данные, то метрики не сильно увеличатся относительно первой модели.

6. Заключение. В работе описан процесс создания синтетических данных пользовательской активности на основе вопросно-ответной выборки. В рамках численных экспериментов удалось получить модель, которая на тестовых данных из похожего домена (алгебраические задачи) показывает сравнительные результаты относительно модели, которая была обучена на реальных данных. При чем эти данные были представлены в намного большем количестве, чем сгенерированные. Также не стоить забывать о начальной инструкции, предоставляемой модели, так как в численном эксперименте при попытке добавить примеры с вопросами и ответами предсказательная способность модели становилась хуже.

В дальнейшем планируется провести аналогичные эксперименты в рамках обучения нейросетевых архитектур, которые требуют более богатую на признаки выборку, добавить больше возможностей для улучшения инструкции для моделей, повторить эксперименты с моделями для русского языка, а также на основе полученных синтетических данных оценить свойства задач из вопросно-ответной выборки, например, оценить сложность заданий (см. [9]).

Литература

1. Brown T. Mann B. et al. Language models are few-shot learners // Advances in neural information processing systems. 2020. Vol. 33. P. 1877–1901.

2. Abdelrahman G., Wang Q., Nunes B. Knowledge tracing: a survey // ACM Computing Surveys. 2023. Vol. 55. No 11. P. 1–37.

3. Чайкин Г. А. Распределение проблемных тематик между учащимися // Процессы управления и устойчивость. 2023. Т. 10. № 1. С. 490–495.

4. Hendrycks D. Burns C. et al. Measuring mathematical problem solving with the math dataset [Электронный ресурс]: URL: `https://arxiv.org/abs/2103.03874` (дата обращения: 19.03.25).

5. Tonga J. C., Clement B., Oudeyer P. Y. Automatic generation of question hints for mathematics problems using large language models in educational technology [Электронный ресурс]: URL: `https://arxiv.org/abs/2411.03495` (дата обращения: 19.03.25).

6. Corbett A. T., Anderson J. R. Knowledge tracing: Modeling the acquisition of procedural knowledge // User modeling and user-adapted interaction. 1994. Vol. 4. P. 253–278.

7. Badrinath A., Wang F., Pardos Z. pyBKT: an accessible Python library of bayesian knowledge tracing models [Электронный ресурс]: URL: `https://arxiv.org/abs/2105.00385` (дата обращения: 19.03.25).

8. 2012-13 School data with affect [Электронный ресурс]: URL: `https://sites.google.com/site/assistmentsdata/datasets/2012-13-school-data-with-affect` (дата обращения: 19.03.25).

9. Луценко М. М., Шадринцева Н. В. Веса Шепли для заданий педагогического теста // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2017. Т. 13. № 3. С. 300–312.

УДК 519.866.2

Шабалов Н. А., Бойко А. В., Малков А. В.

Применение ML алгоритмов для модели тарификации ОСАГО

Рекомендовано к публикации профессором Смирновым Н. В.

1. Введение. Современный страховой рынок невозможно представить без моторного страхования, значительная часть которого приходится на обязательное страхование автогражданской ответственности (ОСАГО). Высокая степень проникновения транспортных средств во все сферы общества обуславливает важность эффективной системы страхования, обеспечивающей финансовую защиту автовладельцев и третьих лиц. Для устойчивого функционирования страхового рынка критически важна точная оценка рисков, что позволяет страховщикам минимизировать потенциальные убытки и принимать обоснованные стратегические решения.

Традиционные методы оценки рисков в страховании, основанные на статистических моделях, таких как обобщённые линейные модели (GLM) [1, 2], обладают рядом ограничений. Они с трудом обрабатывают большие и сложные наборы данных, требуют постоянного обновления и сложной интерпретации [3]. С развитием методов искусственного интеллекта и увеличением доступности данных страховые компании всё чаще обращаются к методам машинного обучения (ML). Использование ML –алгоритмов позволяет не только повысить точность прогнозирования вероятности наступления страховых случаев, но и учитывать динамически изменяющиеся факторы риска, такие как внедрение новых технологий в автомобилестроении или изменения в законодательном регулировании [1, 4].

Кроме точности прогноза риска, ключевым аспектом работы страховых компаний является формирование справедливой и экономически обоснованной тарифной политики. В условиях жесткой кон-

Шабалов Никита Александрович – магистрант, Новосибирский государственный университет; e-mail: n.shabalov@g.nsu.ru, тел.: +7(923)175-09-75

Бойко Алина Владимировна – аспирант, Санкт-Петербургский государственный университет; e-mail: alina.boyko@mail.ru, тел.: +7(911)992-30-87

Малков Андрей Владимировчи – магистрант, Новосибирский государственный университет; e-mail: lozelko1@yandex.ru, тел.: +7(903)068-92-60

куренции и государственного регулирования на рынке ОСАГО важно предлагать страхователям конкурентоспособные тарифы, обеспечивающие баланс между прибыльностью страховой компании и доступностью страховых полисов для клиентов. Машинное обучение открывает возможности для персонализированной тарификации, при которой страховая премия формируется на основе индивидуального риска страхователя [1].

В данной работе рассматриваются подходы к моделированию рисков и тарификации физических лиц по ОСАГО с применением современных алгоритмов машинного обучения.

2. Постановка задачи. Расчет страховой премии ОСАГО в России регулируется Центральным Банком (ЦБ) РФ и основывается на установленных тарифах, которые включают базовый тариф (ТБ) и корректирующие коэффициенты:

$$\text{Премия ОСАГО} = \text{ТБ} * \text{КТ} * \text{КБМ} * \text{КВС} * \text{КО} * \text{КС} * \text{КМ}.$$

Базовый тариф определяется страховой компанией в зависимости от внутренней политики и набора факторов, которые учитываются при оценке риска (в заданных границах), а корректирующие коэффициенты зависят от параметров страхования (территории преимущественного использования (КТ), безаварийности езды водителей (КБМ), стажа и возраста водителей (КВС), ограничения водителей (КО), сезонности использования (КС), мощности автомобиля (КМ)).

Ключевой задачей страховщика является прогнозирование риска на период использования (обычно 1 год), связанного с конкретным страхователем, и определение соответствующего ТБ. Для этого используется анализ исторических данных о страховых случаях. В традиционных методах актуарных расчетов применяются статистические модели, такие как GLM, однако они обладают ограниченной предсказательной способностью и не всегда учитывают нелинейные зависимости.

Современные методы машинного обучения позволяют существенно улучшить точность прогнозирования страховых случаев и динамически адаптировать модели к изменениям в данных. В данном исследовании рассматриваются алгоритмы градиентного бустинга для предсказания вероятности наступления страхового события.

При прогнозировании страховых выплат применяются два подхода [1, 5, 6]:

- Твиди – регрессия (Tweedie – regression). Данный метод основывается на распределении Твиди, которое охватывает широкий спектр возможных значений страховых выплат, включая случаи с нулевыми и положительными значениями. Это делает его удобным для моделирования страховых данных, поскольку страховые убытки часто представляют собой разреженные данные с преобладанием нулевых значений;

- частота – серьёзность (frequency – severity). Данный подход разделяет прогнозирование риска на два этапа:

  $$\text{Частота убытков} * \text{Средний убыток} = \text{Суммарный убыток}.$$

  Сначала оценивается частота наступления страховых случаев с помощью распределения Пуассона, а затем моделируется серьёзность ущерба с использованием гамма – распределения. Этот подход обеспечивает более детализированное моделирование страхового риска.

В рамках проекта использован подход на основе частоты – серьёзности, поскольку он обеспечивает гибкость при настройке бизнес – логики.

- традиционное урегулирование – когда клиент является виновником ДТП, его страховая компания компенсирует ущерб пострадавшим в соответствии с условиями ОСАГО;

- ответственное урегулирование – когда клиент является виновником ДТП, однако пострадавшая сторона обращается в свою компанию, которая возмещает ущерб от компании виновника, используя принцип «обратной суброгации»;

- прямое возмещение убытков (ПВУ) применяется, когда клиент является пострадавшей стороной. Страховая компания пострадавшего выплачивает компенсацию, затем возмещает расходы у страховщика виновника, также по принципу «обратной суброгации» [7, 8].

Для ситуаций, когда клиент является виновником и пострадавшим, строятся модели предсказания убытка, что в сумме составляет четыре ML-модели.

Рассмотрим подробнее архитектурную схему процесса ценообразования (рис. 1).

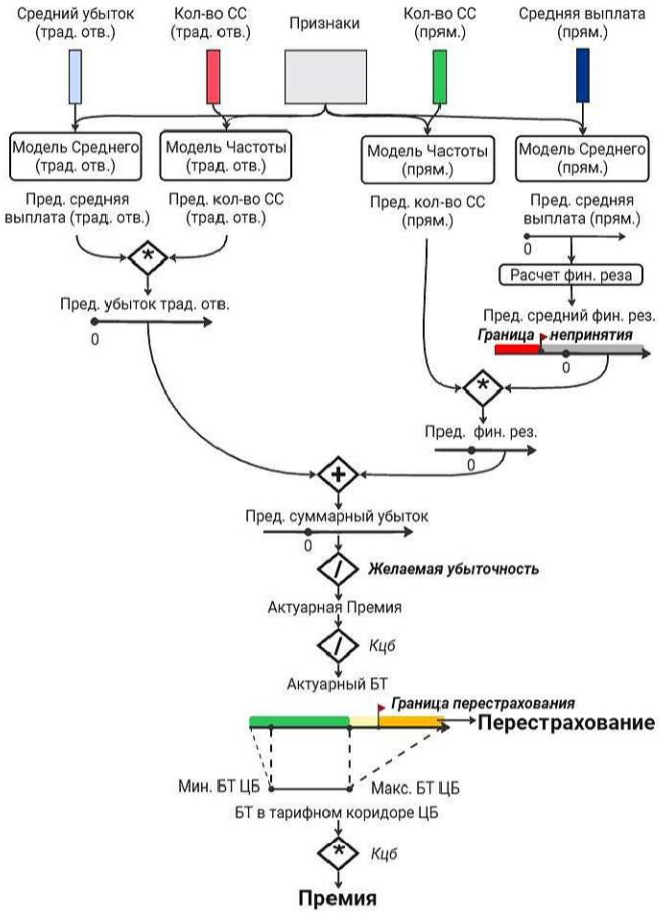


Рис. 1: Общая схема ценообразования

Архитектура решения состоит из двух основных моделей с подходом частота – серьёзность: первая модель предсказывает традиционное и ответственное урегулирование убытков, вторая – убытки при прямом урегулировании. С учётом использования исторических дан-

ных прогнозируемые значения средних убытков дополнительно корректируются на основе ожидаемых темпов инфляции. В рамках прямого урегулирования важен процесс обратной суброгации [9], способный вызвать дополнительную разницу между расчетами и фактическими убытками. Для минимизации таких рисков устанавливается граница непринятия риска, определяющая порог, при котором полис не рекомендуется страховать. На следующем этапе происходит суммирование предсказанных убытков для традиционного, ответственного и прямого урегулирования. Это значение представляет собой предсказанный суммарный риск по конкретному полису. Актуарная страховая премия полиса определяется на основе прогнозируемого уровня риска, нормированного с использованием соответствующего коэффициента. Для получения актуарного ТБ прогноз риска корректируется на поправочные коэффициенты ЦБ [9]. После того как актуарный ТБ полиса вычислен, добавляется граница перестрахования. Все риски, которые превышают установленную границу, попадают в перестраховочный пул и к ним применяется максимальный ТБ. По правилам перестрахования 23% страховой премии остается страховщику, остальная сумма распределяется между участниками перестрахования. Убытки распределяются пропорционально доле рынка страховщика. После применения перестрахования ТБ корректируется с учётом границ, установленных ЦБ.

Данные для обучения моделей собираются из внутренних систем страховой компании и официальных источников, таких как РСА [8] и Банк России. Для анализа используется информация о страхователях, собственниках, водителях и автомобиле. В качестве временного диапазона выбран период с 2019 по 2025 год, что позволяет учесть долгосрочные тренды и сезонные колебания. Для повышения качества моделей проводится предварительная обработка данных, включающая исключение полисов, связанных с мошенничеством и полисов юридических лиц, поскольку они имеют другую бизнес – логику. Кроме того, при моделировании среднего убытка из выборки исключаются полисы без убытков. Категориальные переменные кодируются с помощью целевого кодирования (target encoding).

Целевыми переменными являются количество страховых случаев (CC) и средняя сумма выплат. В обоих случаях используется регрессия. Для прогнозирования всех целевых переменных применён градиентный бустинг на основе решающих деревьев в реализации XGBoost, выбранный за высокую эффективность в задачах регрес-

сии и классификации благодаря методам оптимизации и регуляризации. Данные модели показывают высокую эффективность в страховании [10, 11], особенно при обработке разреженных данных и учёте сложных нелинейных зависимостей.

Для прогнозирования частоты убытков ОСАГО используется XGBRegressor с *objective* = *"count:poisson"*, минимизирующий функцию потерь Пуассона. Для средней суммы ущерба применяется XGBRegressor с *objective* = *"reg:gamma"*, минимизирующий функцию потерь гамма – распределения.

Для оценки качества моделей используются:

- нормализованный коэффициент Джини, который измеряет способность модели ранжировать риски, отражая точность разделения высоко– и низкорисковых полисов

$$G = \frac{2AUC - 1}{AUC_{\max} - AUC_{\text{random}}},$$

где AUC – площадь под ROC – кривой модели, $AUC_{\max}$ – теоретически возможный максимум, $AUC_{\text{random}} = 0{,}5$ для случайной модели;

- средняя абсолютная ошибка, которая показывает среднее отклонение предсказаний от фактических значений

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n} |y_i - \widehat{y}_i|,$$

где y_i – фактические значения, $\widehat{y}_i$ – предсказанные моделью, n – количество наблюдений;

- среднеквадратичная ошибка, которая соответствует квадратному корню из среднего квадрата разницы между целевым значением и значением, предсказанным моделью

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2},$$

где y_i – фактические значения, $\widehat{y}_i$ – предсказанные моделью, n – количество наблюдений.

3. Результаты оценки моделей. Для оценки производительности моделей проведена кросс – валидация на пяти подвыборках и тестирование на отложенной выборке. Датасет охватывает период 01.01.2019–01.03.2025. Для обучения и валидации использованы данные 01.01.2020–01.10.2023, финальное тестирование проведено на отложенной выборке 01.10.2023–01.03.2024. В таблицах 1, 2 представлены значения MAE, RMSE и G для предсказания частоты и серьёзности убытков моделей.

Таблица 1: Результаты моделирования: традиционное и ответсвенное урегулирование

Выборка	Частота			Серьёзность			Убыток		
	MAE	RMSE	G	MAE	RMSE	G	MAE	RMSE	G
Train	0,080	0,232	0,496	52 242	82 005	0,567	6 267	33 183	0,555
Val	0,081	0,237	0,445	59 601	92 104	0,142	6 267	33 162	0,555
Test	0,090	0,234	0,312	56 828	91 505	0,149	7 199	30 495	**0,304**

Таблица 2: Результаты моделирования: прямое урегулирование

Выборка	Частота			Серьёзность			Убыток		
	MAE	RMSE	G	MAE	RMSE	G	MAE	RMSE	G
Train	0,059	0,181	0,569	42 453	69 486	0,694	4 058	21 512	0,666
Val	0,059	0,182	0,517	51 569	83 854	0,412	4 058	21 511	0,666
Test	0,078	0,204	0,361	55 294	85 665	0,271	5 818	23 647	**0,364**

Высокие значения G подтверждают точность модели в предсказаниях частоты и серьезности страховых убытков.

В таблицах 3, 4 представлены важности признаков для прогнозирования частоты и серьёзности убытков. Наиболее значимыми являются данные об автомобиле (модель, тип), местоположении (регион, город, тип населённого пункта) и водителях (количество, возраст, стаж, пол).

Кривая предсказанное/измеренное (рис. 2) показывает высокую степень согласованности предсказанных значений убытков.

Таблица 3: Значимость признаков: традиционное и ответсвенное урегулирование

Частота		**Серьзность**	
Название	**Важность, усл. ед.**	**Название**	**Важность, усл. ед.**
policy_coefficient_kbm	0,23	insurant_city	0,10
vehicle_type	0,08	is_policy_full_period_use_year	0,08
vehicle_count_dtp	0,08	vehicle_make	0,08
owner_city	0,07	policy_region_year_expences_ratio	0,07
policy_coefficient_ko	0,07	insurant_locality_type	0,07
insurant_region	0,05	drivers_age_max	0,07
is_vehicle_commercial	0,05	vehicle_ownership_years	0,07
drivers_experience_max	0,04	drivers_count	0,07
is_policy_full_period_use_year	0,04	policy_coefficient_kbm	0,07
drivers_age_min	0,04	policy_chain_number_by_insurant	0,06
drivers_count	0,03	vehicle_count_mean_damage_points	0,06
policy_coefficient_km	0,03	vehicle_manufacture_type_original	0,06
vehicle_model	0,03	is_vehicle_commercial	0,06
owner_age	0,03	vehicle_count_medium_dtp	0,06
vehicle_power_kwt	0,03		
drivers_gender_male_count	0,02		
vehicle_age	0,02		
vehicle_ownership_years	0,02		
vehicle_count_mean_damage_points	0,02		

Таблица 4: Значимость признаков: прямое урегулирование

Частота		**Серьёзность**	
Название	**Важность, усл. ед.**	**Название**	**Важность, усл. ед.**
vehicle_model	0,20	vehicle_model	0,23
policy_coefficient_general	0,15	is_policy_full_period_use_year	0,11
vehicle_category	0,07	mark_density	0,10
owner_city	0,07	vehicle_power_hp	0,07
drivers_age_max	0,06	insurant_city	0,07
vehicle_count_dtp	0,06	vehicle_age	0,07
vehicle_manufacture_type	0,05	insurant_locality_type	0,06
insurant_locality_type	0,05	is_vehicle_commercial	0,06
insurant_region	0,05	policy_region_year_severity	0,06
policy_chain_number_by_insurant	0,04	drivers_age_max	0,06
insurant_age	0,04	policy_coefficient_kbm	0,05
vehicle_make	0,04	policy_coefficient_km	0,05
vehicle_manufacture_year	0,03		
model_density	0,03		
policy_region_year_expences_ratio	0,02		
vehicle_make_risk	0,02		
drivers_gender_male_count	0,02		



Рис. 2: Сравнение прогноза убытка

4. Выводы. В исследовании рассмотрены методы машинного обучения для прогнозирования рисков и ценообразования ОСАГО. Предложенная модель показала высокую точность в прогнозе вероятности наступления страховых случаев и размеров убытков. Специализированные функции потерь для пуассоновского и гамма–распределений обеспечили корректную работу модели с разреженными данными.

Основные выводы исследования:

- автоматизированные ML – решения помогают гибко управлять рисками и корректировать тарифы с учётом рыночных изменений;
- внедрение ML – системы повысило точность оценки рисков и снизило убыточность при сохранении конкурентоспособности тарифов.

Внедрение модели в промышленную эксплуатацию в рамках страхового портфеля компании позволило снизить убыточность с 0,84 до 0,72 при одновременном двукратном увеличении объёма портфеля. Это подтверждает не только эффективность модели, но и её практическую применимость в условиях реального страхового бизнеса.

В дальнейшем предполагается расширение исследования за счёт более сложных ансамблевых методов и интеграции интерпретируемых моделей для повышения прозрачности решений.

Литература

1. Henckaerts R., Cote M., Antonio K., Verbelen R. Boosting insights in insurance tariff plans with tree – based machine learning methods // North American Actuarial Journal. 2021. Vol. 25. No 2. P. 1–34.
2. Kloese S. GLM Concepts in EMBLEM // ACAS. 2020. Vol. 25. No 8. P. 1–29.
3. Burka D., Kovacs L., Szepesvary L. Modelling MTPL insurance claim events: can machine learning methods overperform the traditional GLM approach? // Hungarian statistical review. 2021. Vol. 4. No 2. P. 34–69.
4. Hanafy M., Ming R. Machine learning approaches for auto insurance Big Data // Risks. 2021. Vol. 9. No 2. P. 1–23.
5. Spilbergs A., Fomins A., Krastins M. Multivariate modeling of motor third party liability insurance claims // European Journal of Business Science and Technology. 2022. Vol. 8. No 1. P. 5–18.
6. Tweedie regression insurance claims [Электронный ресурс]: URL:https://scikit-learn.org/stable/auto_examples/linear_model/plot_tweedie_regression_insurance_claims (дата обращения: 15.03.24).
7. Бойко А. В., Фирсов А. Н. Нахождение равновесия по Нэшу для одной актуарной задачи // Системный анализ в проектировании и управлении. 2020. P. 131–141.
8. Russian association of motor insurers [Электронный ресурс]: URL: https//www.autoins.ru/en/osago (дата обращения: 20.02.2025).
9. Мак Т. Математика рискового страхования. М.: Изд – во Олимп – Бизнес, 2005. 412 с.
10. Tree – Based machine learning for insurance pricing [Электронный ресурс]: URL:https://www.youtube.com/watch?v=z1AlNGXGz9A&ab_channel=RConsortium (дата обращения: 15.03.24).
11. Insurance risk pricing with GLM, GAM and XGBoost [Электронный ресурс]: URL:https://www.youtube.com/watch?v=sOyMLB1SsFk&t=569s&ab_channel=MatthewEvans (дата обращения: 15.03.24).

УДК 004.514, 004.89

Шайдуров В. Д., Корхов В. В., Половков В. В., Пен Е. А., Карельский К. В.

Разработка системы анализа и картирования твердых полезных ископаемых морского дна с использованием искусственного интеллекта

1. Введение. С развитием технологий искусственного интеллекта (ИИ) открываются новые возможности для автоматизации процессов анализа и картирования геологических объектов на морском дне. В работе представлена разработка программно-аппаратного комплекса OpenSeaGeo, предназначенного для картирования твердых полезных ископаемых (ТПИ) морского дна с использованием технологий ИИ.

2. Постановка задачи. Разработать программное обеспечение, способное автоматически распознавать геологические объекты на основе обработки фото- и видео- данных, полученных аппаратной частью в ходе придонного фототелевизионного профилирования океанического дна. Анализ данных подразумевает распознавание геологических объектов и вычисление их характеристик с использованием технологий ИИ, а также нанесение этих объектов на картографическую основу с указанием их геопространственного распределения.

3. Геологические объекты. Геологические объекты для распознавания на видеопотоке включают две группы: железомарганцевые конкреции (ЖМК) и глубоководные полиметаллические сульфиды (ГПС).

Шайдуров Вадим Дмитриевич – студент, Санкт-Петербургский государственный университет; e-mail: st102573@student.spbu.ru, тел.: +7(910)085-05-37

Корхов Владимир Владиславович – доцент, Санкт-Петербургский государственный университет; e-mail: v.korkhov@spbu.ru, тел.: +7(921)947-74-00

Половков Вячеслав Владимирович – доцент, Санкт-Петербургский государственный университет; e-mail: polovkov@gmail.com, тел.: +7(921)947-74-00

Пен Евгений Александрович – исполнительный директор, ООО «БиоГеоХаб»; e-mail: evgeniypen@gmail.com, тел.: +7(921)947-74-00

Карельский Кирилл Вадимович – разработчик, ООО «БиоГеоХаб»; e-mail: karelskykirill@gmail.com, тел.: +7(921)947-74-00

Работа выполнена при поддержке СПбГУ и ООО «Биогеохаб»

В результате картирования ЖМК должны быть распознаны покровы осадочных отложений, выходы коренных пород океанического дна, а также отдельные конкреции с вычислением их видимой плотности залегания, выраженной в процентах от площади кадра.

В результате картирования ГПС должны быть распознаны классы геологических объектов:

- гидротермальные образования: окисленные сульфидные руды, металлоносные осадки, трубные комплексы, гидротермальная фауна;
- геологическое строение океанического дна: редкие глыбы, присыпанные коренники, сплошные осадки, сплошные базальты (текучие покровы, подушечные лавы), дробленые базальты, трещиноватость;
- геоморфологическое строение океанического дна: плоский (пологий) рельеф, расчлененный рельеф, уступы.

4. Организация базы данных. Система работает с двумя основными типами данных:

1. Видеоданные. Источником может быть как реальный видеопоток (например, трансляция с подводной камеры), так и заранее записанные видеофайлы. Видео используется для визуального контроля и анализа объектов с помощью машинного обучения. В процессе анализа выделяются кадры, на которых обнаружены объекты выбранных классов.
2. Географические данные. Данные о местоположении, получаемые от GPS-трекера или загружаемые вручную. Эти данные включают координаты судна или подводного аппарата, а также временные метки для синхронизации с видеопотоком. Географическая информация используется для визуализации на карте: отображения маршрутов, положения аппарата и результатов распознавания объектов.

Таким образом, система оперирует видеоданными и географической привязкой, которые вместе позволяют детектировать объекты и визуализировать их расположение на карте.

Для эффективного хранения и обработки данных океанического дна, была разработана схема базы данных (рис. 1), которая структурирована для обеспечения удобного доступа к информации о видео, геологическим объектам и их характеристикам.

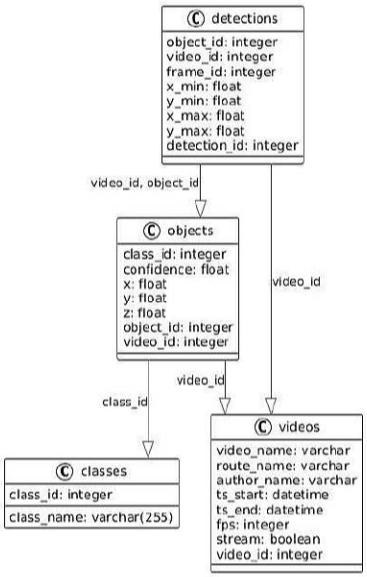


Рис. 1. Блок-схема БД

База данных (БД) включает в себя следующие таблицы:

- таблица «Видео» содержит метаданные о видео, такие как уникальный идентификатор видео, название, маршрут, автор (если над проектом работает несколько человек, необходимо знать, кто добавил видео файл), временные метки начала и окончания записи, количество кадров в секунду и флаг, который указывает на потоковый характер видео;
- таблица «Классы» включает информацию о классах геологических объектов, таких как ЖМК и ГПС. Каждый класс имеет уникальный идентификатор и название;
- таблица «Объекты» хранит данные о распознанных объектах, включая их уникальный идентификатор, идентификатор ви-

део, к которому они относятся, класс объекта, уровень уверенности в распознавании и пространственные координаты;

- таблица «Детекции» хранит координаты маски распознавания объекта на видео – рамка на видеокадре, в которую вписан объект. Записи хранятся по составному первичному ключу: идентификатор видео, номер кадра, идентификатор объекта.

Такая организация БД позволяет эффективно управлять большими объемами данных, обеспечивая возможность быстрого поиска, анализа и визуализации геологических объектов.

5. Техническая реализация системы обнаружения подводных объектов. В основе системы лежит современная архитектура YOLOv7, специально адаптированная для работы в сложных подводных условиях. Эта модель сочетает высокую скорость обработки (до 60 кадров в секунду на видеокарте уровня NVIDIA GeForce RTX 3090) с впечатляющей точностью обнаружения даже при низком контрасте и сложном фоне, характерных для подводных съемок.

Процесс обучения модели был многоэтапным. Изначально производилась работа с небольшим датасетом всего из 250 изображений, но постепенно он был расширен до 3000 снимков. Чтобы модель лучше справлялась с разнообразием подводных условий, были применены несколько мощных техник аугментации данных:

- мозаичное преобразование – комбинирование нескольких изображений в одном кадре;
- цветовые искажения – имитация изменений освещения и мутности воды;
- геометрические трансформации – повороты, масштабирование, искажения перспективы;
- добавление специально сгенерированных сложных примеров.

Для ускорения работы системы была проведена существенная оптимизация:

- перевод модели в ONNX-формат, что дало прирост скорости на 15–20%;
- реализация аппаратного ускорение декодирования видео через NVIDIA Video Codec SDK;

- для работы в реальном времени используется смешанная точность вычисления на GPU NVIDIA, что позволяет обрабатывать до 60 кадров в секунду с задержкой менее 17 мс на кадр. Это обеспечивает плавную работу даже при обработке потокового видео с подводных аппаратов.
- внедрение пайплайна предобработки с батчингом и ресайзингом изображения до 640×640 пикселей;

Обработка видеопотока построена по многоступенчатой схеме:

1. Детекция объектов с помощью YOLOv7.
2. Трекинг через алгоритм SORT (венгерский алгоритм с фильтром Калмана).
3. Точная сегментация через Segment Anything Model (SAM).
4. Анализ пространственных характеристик объектов.

Модель SAM реализована следующим образом: получает ограничивающие прямоугольники от YOLO и на их основе строит точные до пикселя маски объектов. Это позволяет сочетать скорость детекции с точностью сегментации.

Система развернута в Docker-контейнере с REST API интерфейсом, что обеспечивает простую интеграцию с существующей инфраструктурой и удобное обновление модели.

6. Техническая реализация интерфейса. Приложение разработано с использованием PySide6 на Python 3.12, что делает его кроссплатформенным, т. е. способным работать на различных операционных системах, включая Windows, macOS и Linux.

Для хранения данных приложение использует SQLite – легковесную и встроенную в Python базу данных, что упрощает развертывание приложения, но накладывает определенные ограничения, связанные с параллельным доступом на чтение и запись в БД. При этом управление БД реализовано в независимом компоненте, что позволяет в будущем легко заменить SQLite на любую другую СУБД (например, PostgreSQL или MySQL) без необходимости переписывать основную логику приложения.

На рис. 2 представлена часть интерфейса приложения для анализа видеоданных. В центре располагается видео с нанесенными на него детекциями в ходе работы нейронной сети.

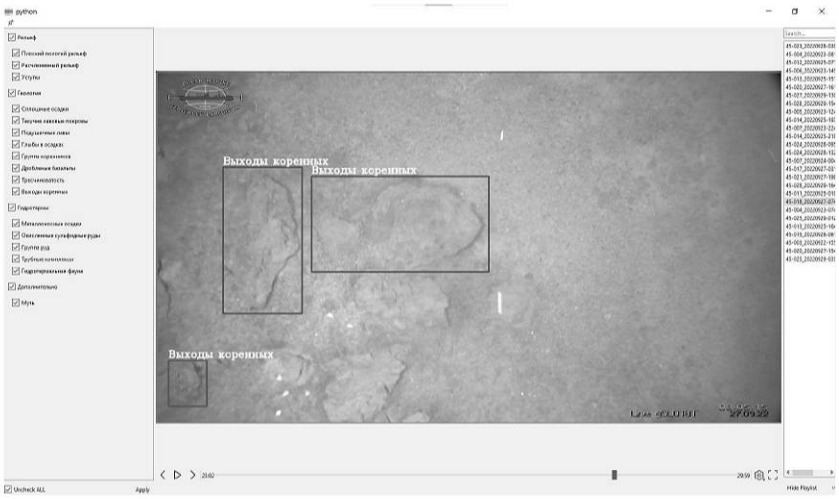


Рис. 2. Окно для отображения детекций на видео

6.1. Архитектура. Архитектура приложения модульная и слоистая, с четким разделением ответственности между компонентами (рис. 3):

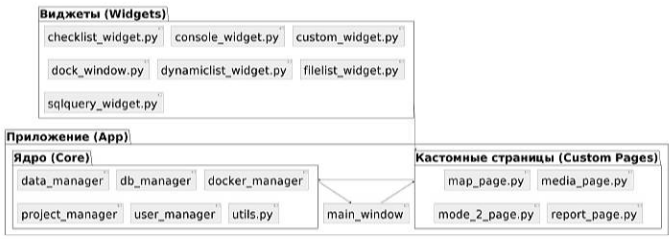


Рис. 3. Архитектура

- виджеты (Widgets) – независимые модули, которые используются для построения пользовательского интерфейса;
- ядро (Core) – это центральная логика приложения. Включает модули для управления данными, БД, Docker-контейнерами, сессией пользователя и всем проектом;

- кастомные страницы (Custom Pages) – это страницы интерфейса для специфических задач, таких как работа с картами, медиа-контентом (видео, изображения), отчетами и т. д.

Такая архитектура позволяет легко расширять приложение, добавляя новые виджеты, страницы или менеджеры, не нарушая существующую структуру.

6.2. Внутреннее устройство. Все операции с чтением и записью данных выполняются в отдельных потоках с использованием QThread из PySide6, что позволяет не блокировать основное окно приложения. Такой подход обеспечивает высокую отзывчивость и производительность системы, но требует особой осторожности, некорректно завершенные потоки способны вызвать утечку в памяти. Благодаря разграничению ответственности, только ядро (рис. 4) отвечает за управление запущенных потоков в приложении, таким образом любое изменение данных в БД порождает сначала сигнал в ядре, а только потом сигнал передается на обновление затронутых изменением страниц.

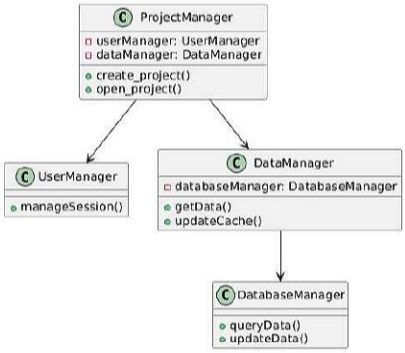


Рис. 4. Ядро

ProjectManager отвечает за инициализацию проекта. Когда выбран существующий проект или создан новый, ProjectManager инициализирует UserManager, который отвечает за права пользователя на изменение проекта и его окружения (в ПО предусмотрено разделение ролей, так как предполагается совместная работа над одним проектом). Если проект новый, то создавший его пользова-

тель имеет абсолютные права. Также ProjectManager инициализирует DataManager, который кэширует необходимые данные из БД проекта и поддерживает их актуальность. DatabaseManager – менеджер запросов в БД. Разделение ответственности между DataManager и DatabaseManager позволяет оптимизировать работу с данными, снижая нагрузку на БД и ускоряя выполнение операций.

Ядро выступает связующем звеном между обособленными виджатами, которые представляют собой самостоятельные компоненты, выполняющие конкретную задачу. Благодаря такой модульности возможно создавать страницы любой сложности и функциональности. Страницы можно разбить на два типа:

1. Страницы представления (рис. 5) только отображают (представляют) данные, которые хранятся в БД. У таких страниц нет прав на запись в БД, а все данные поступают из DataManager, который следит за их актуальностью.

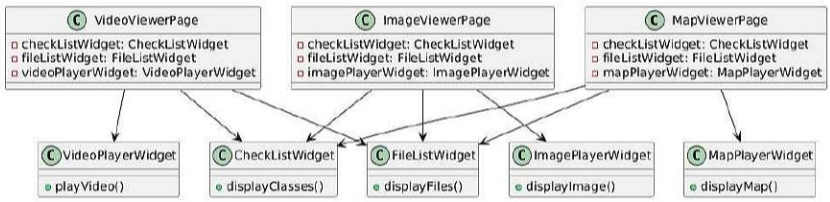


Рис. 5. Страницы представления

2. Служебные страницы (рис. 6), в отличие от страниц представления, могут как считывать, так и записывать данные в БД.

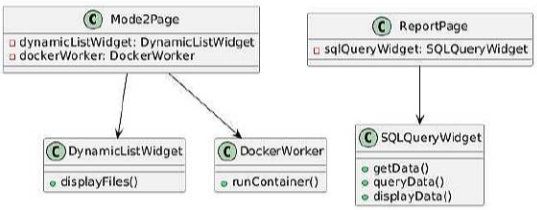


Рис. 6. Служебные страницы

7. Заключение. Разрабатываемая система OpenSeaGeo представляет собой мощный инструмент для автоматизации процессов анализа и картирования твердых полезных ископаемых морского дна. Использование технологий ИИ позволяет значительно ускорить обработку данных, повысить точность распознавания геологических объектов и снизить затраты на проведение геологоразведочных работ. Внедрение такой системы в практику морских исследований открывает новые возможности для изучения и освоения минеральных ресурсов мирового океана.

Литература

1. Ричардс М., Форд Н. Фундаментальный подход к программной архитектуре: паттерны, свойства, проверенные методы. СПб.: Питер, 2023. 400 с.

2. Ричардс М., Форд Н., Садаладж П., Дехгани Ж. Современный подход к программной архитектуре: сложные компромиссы. СПб.: Питер, 2023. 480 с.

3. Шилдс У. SQL: быстрое погружение. СПб.: Питер, 2022. 222 с.

4. Атлас морфологических типов железомарганцевых конкреций Тихого океана [Электронный ресурс]: URL:https://www.geokniga.org/bookfiles/geokniga-1985atlasfe-mnconcretions.pdf (дата обращения: 20.03.2025).

5. Горбачев В. А., Криворотов И. А., Маркелов А. О. и др. Семантическая сегментация спутниковых снимков аэропортов с помощью сверточных нейронных сетей // Компьютерная оптика. М.: МФТИ, 2020. Т. 44. № 4. С. 636–645.

6. Минченков М. В., Хельвас А. В., Юрин Д. В. Алгоритм автоматической сегментации изображений для систем технического зрения // Обработка информации и моделирование. М.: МФТИ, 2002. 80 с.

7. Qt for Python documentation [Электронный ресурс]: URL:https://doc.qt.io/qtforpython-6 (дата обращения: 20.03.2025).

УДК 004.912

Юрьевцева П. Н.

Исследование чувствительности LLM к психоэмоциональным параметрам пользователей на основе профилирования синтетических персон

Рекомендовано к публикации доцентом Митрофановой О. А.

1. Введение. В современном мире многие социально-значимые задачи решаются с применением моделей искусственного интеллекта (ИИ). Для обеспечения безопасности и эффективности информационного обмена цифровое общество предъявляет высокие требования к разрабатываемым ресурсам [1]. В частности, широкое применение находят большие языковые модели (LLM), способные генерировать связные тексты разных типов и встраиваемые в системы разговорного ИИ. Использование LLM в чат-ботах обнаруживает проблему персонификации языковых моделей. Обучение LLM на сверхбольших корпусах класса WaC (Web as Corpus), фиксирующих тематику, жанрово-стилистическое разнообразие текстов и коммуникативные практики языкового коллектива, приводит к тому, что признаки, формирующие социолингвистические и психологические профили отдельных авторов, могут стираться. Эта особенность LLM расширяет сферу их безопасного применения в нейтральных коммуникативных ситуациях, но, с другой стороны, предполагает дополнительную адаптацию к индивидуальным особенностям и психоэмоциональному состоянию пользователей чат-ботов [2], что актуально в разработке инклюзивных ИИ-помощников для отдельных групп говорящих (в частности, для лиц с РАС, синдромом Аспергера, СДВГ и т. д.). В UI/UX-дизайне разработаны подходы к тестированию интерфейсов, опирающиеся на синтетические персоны, которые воспроизводят основные характеристики целевой аудитории [3]. Целью исследования является экспериментальная проверка гипотезы о чувствительности LLM к лексическим маркерам эмоций с применением эмоциональных синтетических персон в промпт-инжиниринге [4].

Юрьевцева Полина Николаевна – студент, Санкт-Петербургский государственный университет; e-mail: st097486@student.spbu.ru, тел.: +7(982)904-58-65

2. Экспериментальный дизайн. Проверка гипотезы о чувствительности LLM к психоэмоциональному состоянию коммуникантов потребовала проведения двух серий экспериментов.

2.1. Эксперимент 1. Цель эксперимента 1 заключается в оценке способности LLM распознавать эмоциональную лексику по сравнению с носителями языка. В центре внимания находятся русскоязычные имена эмоций, список которых был сформирован на основе семантической разметки НКРЯ [5] и содержит 105 частотных существительных. Из проанализированных подходов к систематизации эмоций была выбрана двумерная модель Дж. Рассела, которая представляет эмоции как точки на плоскости, определяемой двумя осями: тональностью (от отрицательной к положительной) и интенсивностью (от спокойствия к возбуждению) [6]. Был проведен опрос респондентов, которым в качестве стимулов были предъявлены имена эмоций и предлагалось оценить значение каждого из стимулов по двум шкалам от 1 до 10. Ответы шести респондентов были усреднены и представлены в виде множества точек в декартовой системе координат. Методом zero-shot-промптинга для данного набора стимулов были получены ответы от LLM, в качестве которой была использована YandexGPT Pro 5. Результаты отражены на рис. 1.

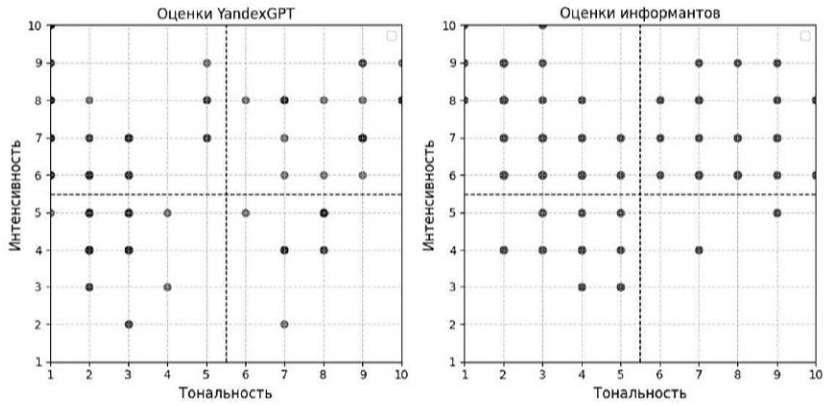


Рис. 1. Представления оценок в декартовой системе координат

В ходе анализа ответов было установлено, что респонденты хуже распознают названия слабоинтенсивных положительных эмоций, к

которым были отнесены только спокойствие и жалость, и лучше – названия интенсивных отрицательных эмоций, например, недоумение, горечь, раздражение и др. Различия в распределении эмоциональных значений подтверждаются низкими оценками согласованности между усредненными ответами респондентов и LLM по Каппе Коэна

$$\kappa = \frac{p_o - p_e}{1 - p_e},$$

где p_o – доля полного согласия, p_e – вероятность случайного согласия.

Каппа Коэна для оценок по шкале тональности равна 0,192, по шкале интенсивности – 0,136, что соответствует слабой согласованности оценок и подтверждает необходимость проведения эксперимента 2.

2.2. Эксперимент 2. Цель эксперимента состоит в оценке семантической близости текстов ответов LLM, полученных с помощью промптов, в которых содержится информация о контекстных маркерах названий эмоций и об эмоциональном состоянии синтетических персон.

2.3. Данные. В эксперименте 2 список названий эмоций ограничен лексикой, представленной в колесе Р. Плутчика (32 эмоции). В данной статье приведены результаты по восьми базовым эмоциям, характеризуемым наибольшей интенсивностью: гнев, изумление, отвращение, ужас, восхищение, восторг, горе, настороженность. Было использовано четыре промпта с описанием синтетических персон, испытывающих ту или иную эмоцию. Данные персоны различаются по возрасту и полу (ребенок – взрослый, мужчина – женщина). На сформированные zero-shot-запросы было получено 32 ответа модели, которые включают персонализированные эмпатичные рекомендации для проживания эмоций.

2.4. Модель и метрика. Для оценки семантической близости ответов LLM текстовые данные были векторизованы и представлены в виде эмбеддингов с использованием библиотеки Sentence-Transformers и предобученной многоязычной модели, построенной на архитектуре трансформер, ru-en-RoSBERTa [7]. В качестве метрики используется косинусная близость векторов текстов

$$\text{Similarity}(A, B) = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}}.$$

Выбор модели обусловлен ориентированностью модели на русский язык, обучением модели на базовых датасетах для русского языка, а также на общедоступных данных из различных доменных областей, высокой эффективностью для задач NLP, в том числе в задаче оценки семантической близости текстов [8]. Архитектура модели включает токенизатор ruRoBERTa, дополненный токенами из RoBERTa. Для обучения новых векторных представлений токенов модель дообучалась с использованием задачи Masked Language Modeling (MLM), предсказывая слова на месте замаскированного токена [MASK] на основе контекста.

Следуя [3], мы получили описательную статистику в соответствии с персонами (рис. 2) и эмоциями (рис. 3) в виде усредненных значений косинусной близости внутри групп векторизованных текстов, объединенных соответствующими признаками.

2.5. Результаты эксперимента 2.

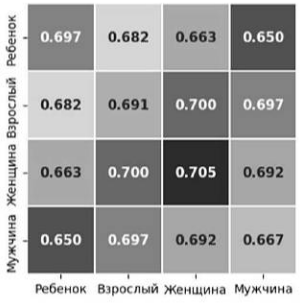


Рис. 2. Описательные статистики для персон

При оценке чувствительности LLM к возрасту и полу коммуникантов значения косинусной близости находятся в интервале (0,65; 0,705). Значения выше для групп векторов текстов, сгенерированных моделью в ответ на промпты для персон «ребенок» и «женщина», ниже – для персон «взрослый» и «мужчина». Наибольшие различия наблюдаются между персоной «ребенок» и остальными персонами, что говорит о дифференциации относительно возраста.

Оценка чувствительности LLM к маркерам эмоций (рис. 3) показала, что значения косинусной близости в группах векторизованных текстов, соотносящихся с целевыми названиями эмоций, находятся в интервале (0,594; 0,934), при этом самые низкие значения косинусной меры соответствуют интенсивным положительным эмоциям (восторг и восхищение), а самые высокие – интенсивным отрицательным (гнев и ужас).

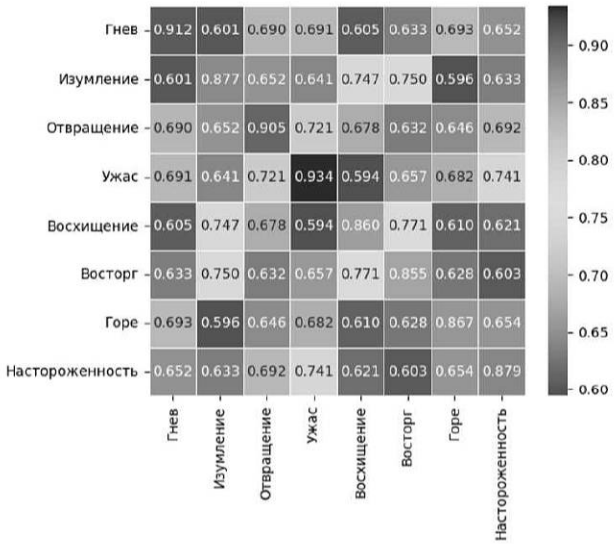


Рис. 3. Описательные статистики для эмоций

3. Заключение. В данной статье представлены результаты экспериментов по определению чувствительности YandexGPT Pro 5 к лексическим маркерам эмоций с привлечением русскоязычных данных. При проведении эксперимента 1 на основе лексикографических источников и корпусных данных сформирован список частотных существительных – названий эмоций в русском языке. В ходе опроса респондентов по разметке эмоциональных значений согласно шкале Дж. Рассела с параллельным тестированием LLM установлена низкая согласованность ответов носителей языка и модели. В результате эксперимента 2 был получен новый датасет, содержащий лексемы – названия эмоций из колеса Р. Плутчика, а также тексты – описания эмоций, сгенерированные LLM на основе промптов, содержащих указания на пол, возраст и эмоциональное состояние говорящего. Полученные описательные статистики подтверждают исходную гипотезу о том, что эмбеддинги LLM способны дифференцировать как социодемографические признаки пользователей (возраст и пол), так и их

психоэмоциональные характеристики. В дальнейшем предполагается привлечение альтернативных способов векторизации, расширение языкового материала и проведение дисперсионного анализа ANOVA на основе описательных статистик.

Литература

1. Андрейченко А. Г., Блеканов И. С. Нейросетевые методы анализа мнений пользователей в дискуссиях социальных сетей // Процессы управления и устойчивость. 2022. Т. 9. № 1. С. 165–174.

2. Lin H., Feng S., Geishauser C. EmoUS: simulating user emotions in task-oriented dialogues [Электронный ресурс]: URL:https://arxiv.org/abs/2306.01579 (дата обращения: 13.11.2024).

3. Bakaev M., Gorovaia S., Mitrofanova O. Who will author the synthetic texts? Evoking multiple personas from Large Language Models to represent users' associative thesauri // Big Data and Cognitive Computing. 2025. Vol. 9. No 2. Art. no 46.

4. Sahoo P., Singh A. K., Saha S. A systematic survey of prompt engineering in Large Language Models: techniques and applications [Электронный ресурс]: URL:https://arxiv.org/abs/2402.07927 (дата обращения: 7.03.2025).

5. Национальный корпус русского языка [Электронный ресурс]: URL:https://ruscorpora.ru/ (дата обращения: 18.02.2025).

6. Овсянникова В. В. К вопросу о классификации эмоций: категориальный и многомерный подходы // Финансовая аналитика: проблемы и решения. 2013. № 37. С. 43–48.

7. Model Card for ru-en-RoSBERTa [Электронный ресурс]: URL:https://huggingface.co/ai-forever/ru-en-RoSBERTa (дата обращения: 10.03.2025).

8. Snegirev A., Tikhonova M., Maksimova A. The Russian-focused embedders' exploration: ruMTEB benchmark and Russian embedding model design [Электронный ресурс]: URL:https://arxiv.org/abs/2408.12503 (дата обращения: 10.03.2025).

UDC 004.8

Fan J., Liu J., Davydenko A. A.

Applying MegaAgent framework for oil and gas extraction optimization

1. Introduction. Oil and gas extraction is a complex process that involves optimization of parameters such as pressure, recovery rate, and equipment wear. Conventional methods typically rely on heuristic or single-intelligent-body systems, which struggle to adapt when faced with dynamic and uncertain conditions in real oil fields. These limitations highlight the need for more intelligent and adaptive optimization methods.

In recent years, Reinforcement Learning (RL), in particular Multi-agent Reinforcement Learning (MARL), based on the MADDPG algorithm, has provided promising solutions to this problem. MADDPG enables multiple intelligences to collaborate and optimize their decisions in real time, improving system efficiency and safety. In this study, MADDPG will be applied in the MegaAgent framework to optimize the oil and gas extraction process and improve extraction efficiency by coordinating, pressure control, and oil extraction intelligence.

2. Oil and gas extraction problem modeling. We formulate the problem as a game.

2.1. State space. The state space of the oil and gas extraction process includes several state variables of the well, such as pressure, flow, and equipment health. The state space can be represented as:

$$S = \{s_1, \ldots, s_n\},$$

where s_i represents the state of the well at a given time [1].

2.2. Action space. Each agent's action space is continuous, such as pressure control (adjusting well pressure) and extraction control (ad-

Fan Jiajie – graduate student, Saint Petersburg State University; e-mail: st122445@student.spbu.ru, phone: +7(963)318-66-59

Liu Jie – graduate student, Saint Petersburg State University; e-mail: st122357@student.spbu.ru, phone: +7(981)855-33-39

Davydenko Aleksandr Aleksandrovich – senior lecturer, Saint Petersburg State University; e-mail: a.davydenko@spbu.ru, phone: +7(911)210-57-92

justing the extraction rate). The action for each agent can be represented as:

$$A_{\text{pressure}} = \{a_1, \ldots, a_m\}, \quad A_{\text{extraction}} = \{a_1, \ldots, a_m\},$$

where a_i represents the action taken by the agent in a given state.

2.3. Reward function. To maximize the oil and gas extraction efficiency and reduce equipment wear, the reward function design is crucial. A typical reward function might be

$$R_t = w_1 \cdot \text{reward_oil}_t + w_2 \cdot \text{reward_pressure}_t + w_3 \cdot \text{reward_wear}_t,$$

where reward_oil_t represents the oil extraction reward, reward_pressure_t represents the pressure control reward, and reward_wear_t represents the equipment wear penalty. The parameters w_1, w_2, and w_3 are weight factors to balance the different impacts.

3. Proposed solution. In this section we describe MADDPG and MegaAgent framework, to make clear how we combine them.

3.1. Multi-Agent Deep Deterministic Policy Gradient. Multi-Agent Deep Deterministic Policy Gradient (MADDPG) is a multi-agent reinforcement learning algorithm that combines Deep Deterministic Policy Gradient (DDPG) with multi-agent collaboration, specifically designed for continuous action spaces. This algorithm uses a centralized training, decentralized execution (CTDE) [2] framework for training multiple agents, enabling them to operate independently while cooperating toward a common objective.

Centralized training means that all agents share a global Q-value and policy network during training and optimize based on global rewards.

Decentralized execution means that each agent makes decisions based on local observations, but their actions are influenced by centralized training.

3.2. Actor and critic networks [3]. Each eagent has two networks: *actor* and *critic*, which act in the following manner.

Actor network generates continuous actions based on the current state. The goal of the Actor network is to maximize the Q-value output by the Critic.

Critic network evaluates the quality of the current state-action pair by outputting a Q-value.

Critic update is made by minimizing the mean squared error between the current Q-value and the target Q-value

$$L(\theta_i) = \frac{1}{S}\sum_j \left(y_j - Q_i^{\mu}(x_j, a_1^j, \ldots, a_N^j)\right)^2,$$

where y_j is the target Q-value computed by the target networks(S represents the number of samples drawn from the experience replay pool).

Actor update is carried out through policy gradient. Specifically, the Actor network aims to maximize the Q value of the Critic network and update its policy to increase long-term rewards:

$$\nabla_{\theta_i} J \approx \frac{1}{S}\sum_j \nabla_{\theta_i}\mu_i(o_i^j)\nabla_{a_i} Q_i^{\mu}(x^j, a_1^j, \ldots, a_N^j)\Big|_{a_i=\mu_i(o_i^j)}.$$

To stabilize the training process, the *target networks* are updated using a *soft update*

$$\theta' \leftarrow \tau\theta + (1-\tau)\theta',$$

where θ represents the current networks parameters, θ' represents the target networks parameters, and τ is the soft update factor.

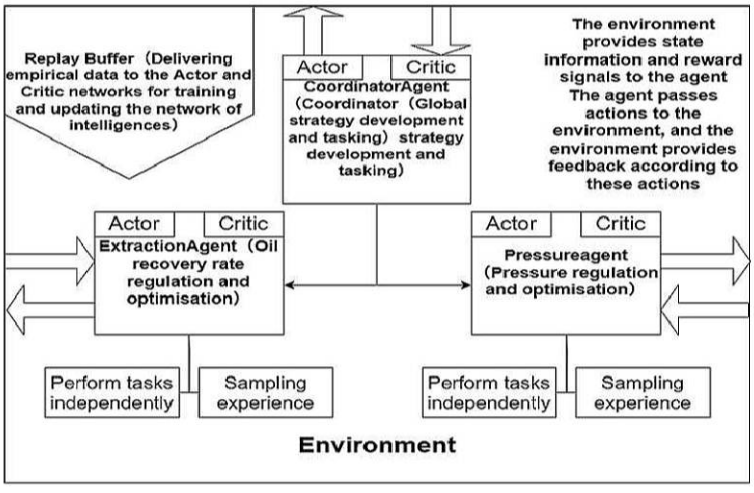


Figure 1. Framework

3.3. MegaAgent framework overview. The MegaAgent framework [4] is designed to optimize the oil and gas extraction process through efficient task decomposition, agent collaboration, and parallel execution. It incorporates several key components and processes as outlined below.

Task decomposition. The framework handles the complexity of oil and gas extraction tasks by adopting hierarchical task decomposition. Tasks are divided into smaller sub-tasks, which are then assigned to different agents. For instance, the coordinator agent assigns tasks, while the pressure control agent and extraction control agent are responsible for controlling well pressure and extraction rates, respectively.

Agent collaboration. Each agent makes decisions based on its local observations. However, through the MegaAgent framework, agents collaborate to optimize the overall extraction process, ensuring that the tasks are executed efficiently and effectively.

Framework architecture. The framework includes various agents, such as the `CoordinatorAgent`, `PressureAgent`, and `ExtractionAgent`. Each agent interacts with the environment and other agents based on its current state. This interaction helps achieve a common goal of optimizing the extraction process. While agents execute tasks independently based on local observations, the training process follows a centralized approach. Centralized training aligns the agents towards a global objective, ensuring coordinated and efficient task execution.

Parallel execution and resource management. The framework uses task decomposition and parallel execution of agents to handle large-scale oil and gas extraction tasks, reducing the time required for task completion. Effective resource management ensures that each agent utilizes computational and storage resources optimally while performing tasks.

Training and evaluation. During the training process, agents accumulate experience through their interactions with the environment. They update their Q-values and policies based on the shared global reward to improve their performance. Periodic evaluations are conducted to assess agent performance, ensuring continuous improvements in coordination and overall efficiency.

4. Results evaluation. We decribe here the experimental environment and the obtained results.

4.1. Experimental environment. Table 1 summarizes key environmental parameters critical to the oil well recovery process, including

maximum pressure, optimum extraction pressure, maximum equipment wear, and initial oil volume. These parameters define the operational limits and optimization objectives, guiding the simulation of oil and gas extraction. The code dynamically adjusts them to optimize extraction efficiency, control equipment wear, and maintain operation within safe pressure ranges.

Table 1. Environmental parameters of oil wells

Parameter Name	Value	Description
Max Pressure	5000	The maximum pressure limit the well can withstand
Optimal Pressure	3000	The optimal working pressure of the well
Max Wear	100	The maximum wear value the equipment can withstand
Max Oil Quantity	10000	The initial oil quantity in the well
Max Steps	200	The maximum number of steps per episode

Table 2 outlines the agents in the MegaAgent framework and their control objectives. The Coordinator Agent formulates global strategies to coordinate other agents, ensuring smooth pressure control and extraction. The Pressure Control Agent regulates the well's pressure within the optimal range for safety and efficiency. The Extraction Control Agent optimizes the oil extraction rate. Together, these agents collaborate to improve the overall efficiency of oil and gas extraction while minimizing equipment wear.

Table 2. Intelligent Body Configuration

Agent	Main Function	Control Objective
Coordinator	Global strategy formulation	Coordinate pressure control and extraction operations
Pressure Control	Pressure regulation	Maintain pressure within the optimal range
Extraction Control	Oil rate control	Optimize oil extraction efficiency

Assessment of indicators:

- Control indicators: pressure error, stability.
- Efficiency indicators: oil recovery, overall efficiency.
- Life indicators: wear rate, cumulative wear.
- Performance indicators: average reward, stability, convergence rate.

4.2. Experimental results. MADDPG outperforms DDPG, PPO, and SAC in oil and gas extraction tasks across all aspects (see Fig. 2). It

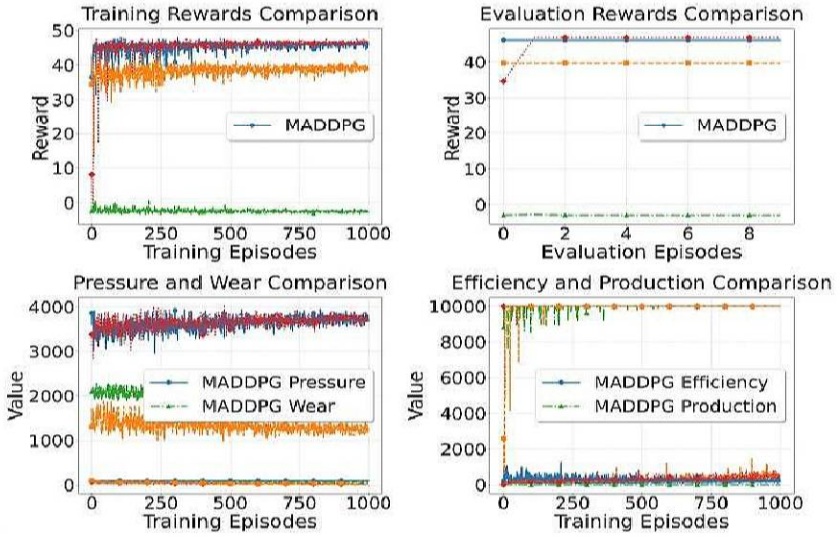


Figure 2. Algorithm comparison

effectively balances extraction efficiency and equipment protection, optimizing task execution. While SAC performs well in some areas, it shows larger fluctuations and less stability than MADDPG. PPO [5] ranks worst in training rewards, and DDPG is the least effective in pressure and wear comparisons. Overall, MADDPG is the most effective algorithm for this task, balancing multi-agent collaboration, oil extraction, pressure control, and equipment wear. Other algorithms may need further adaptation or improvement for similar environments.

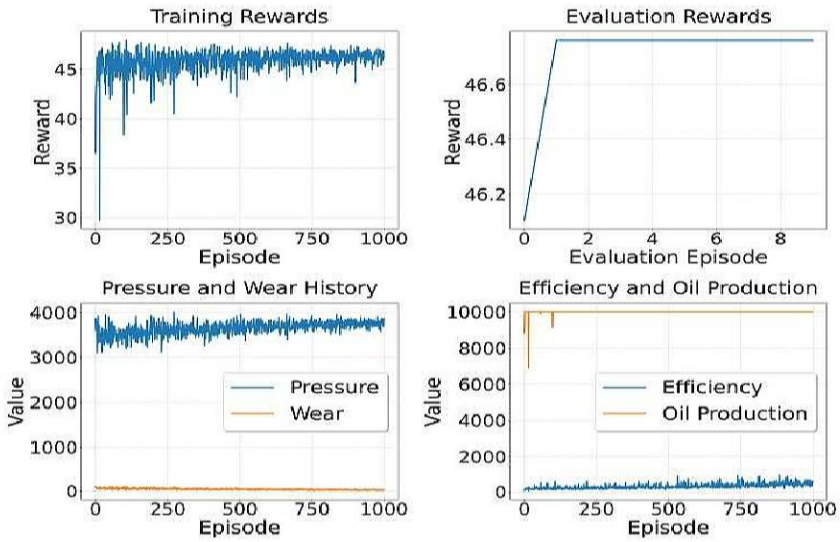


Figure 3. Training analysis

5. Conclusion. This study presents an intelligent oil well control system based on the MegaAgent framework with multi-intelligence reinforcement learning. It achieves precise control of well pressure and optimized oil recovery efficiency through the collaboration of three agents: coordinator, pressure control, and oil recovery control. Experimental results show that the system maintains stable well pressure (3000 ± 500 Pa), improves oil recovery efficiency compared to traditional DDPG, and effectively controls equipment wear. The system converges within 200 training rounds (see Fig. 3), demonstrating strong performance and providing a reliable solution for intelligent oilfield construction.

References

1. Noguer i Alonso M., Njupoun A. M. Game theory and multi-agent reinforcement learning: a mathematical overview // SSRN, Aug 14, 2024. [Web resourse]: `https://ssrn.com/abstract=4926191` (Accessed May 27, 2025).

2. Gronauer S., Diepold K. Multi-agent deep reinforcement learning: a survey // Artificial Intelligence Review. 2022. Vol. 55. P. 895–943.

3. Lowe R., Wu Y., Tamar A. Multi-agent actor-critic for mixed cooperative-competitive environments // Proc. of the 31st International Conference on Neural Information Processing Systems (NIPS'17). 2017. P. 6382–6393.

4. Wang Q., Wang T., Li Q., et al. MegaAgent: a practical framework for autonomous cooperation in large-scale LLM agent systems // arXiv preprint, 2024. https://arxiv.org/abs/2408.09955v2

5. Lillicrap T. P., Hunt J. J., Pritzel A., et al. Continuous control with deep reinforcement learning // arXiv preprint, 2019. https://arxiv.org/abs/1509.02971v6

UDC 519.8

Fan X., Blekanov I. S., Gubar E. A.

Evolutionary approach based on BERTopic and NetLogo to analyse user discussions in social media

1. Introduction. In the era of social media, public engagement with trending topics has driven the complex evolution of information dissemination mechanisms. While current research has made progress in the fields of topic modeling and multi-agent simulation, it often disconnects the two: topic modeling is mostly confined to static feature analysis, lacking quantitative evaluation of dissemination dynamics; multi-agent simulation frequently relies on simplified assumptions and is not sufficiently validated with real social media data. Addressing this limitation, this paper innovatively constructs a BERTopic-NetLogo integrated framework [1], [2], using the "climate change" topic as an empirical case to validate the model's effectiveness, thereby providing new methodological support for the study of social media topic dissemination.

At the methodological design level, BERTopic is used for topic mining: this model generates hierarchical topics by combining pre-trained language models with c-TF-IDF weighting, effectively addressing the issues of LDA [3] in short text modeling. Meanwhile, NetLogo is utilized as a multi-agent simulation tool, its open-source architecture and visual interactive interface providing robust support for accurately simulating the multi-agent dynamic dissemination process in social media. Compared to specialized tools like Altreva Adaptive Modeler [4], NetLogo offers broader applicability, breaking through domain limitations.

The innovation of this paper lies in the integration of topic modeling with multi-agent simulation, achieving a full-process analysis from topic

Fan Xuanping – graduate student, Saint Petersburg State University; e-mail: st115931@student.spbu.ru, phone: +7(812)363-65-37

Blekanov Ivan Stanislavovich – associate professor, Saint Petersburg State University; e-mail: i.blekanov@spbu.ru, phone: +7(812)363-65-37

This research has been supported by the project "Cumulative public opinion and deliberation: Theorizing, research methodologies, and quality assessment of today's deliberative process", ID 128786104, 2025

Gubar Elena Alekseevna – associate professor, Saint Petersburg State University; e-mail: e.gubar@spbu.ru, phone: +7(812)363-65-37

The research was supported by the Russian Science Foundation grant No 24-21-00302

classification to dissemination simulation; by quantitatively evaluating information influence, it provides a new dimension for topic dissemination research; and by validating simulation results with real data, it enhances the model's reliability and practicality.

2. Methods and materials.

2.1. Pipeline. The pipeline of this study is shown in Figure 1. First, data is collected from the Weibo platform using web crawling techniques, and the data is preprocessed. Subsequently, the preprocessed data is fed into the BERTopic model to extract subtopics and their related features. Then, the process moves to post processing, where, after a series of treatments, it provides the input for the subsequent multi-agent simulation. Next, a multi-agent model is built using NetLogo to simulate the dynamic dissemination process of topics within the social network. Finally, the original data distribution and the predicted data distribution are output.

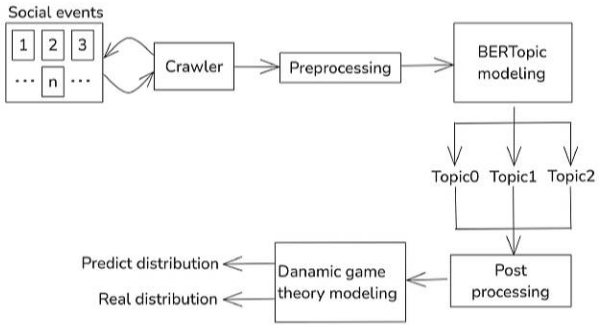


Figure 1. Pipeline

2.2. Post processing module. This module is one of the core components of the process, and its primary task is to transform the subtopic information extracted by BERTopic into input data suitable for multi-agent simulation. The specific steps are as follows:

1. Influence Ranking Calculation: The likes, comments, and reposts of each piece of information are ranked from high to low, and the ranking intervals are divided into 10 levels.
2. The payoff of a single piece of information is calculated using formula (1), which comprehensively considers the influence of the in-

formation across three dimensions: likes, comments, and reposts. The higher the payoff, the greater the influence of the information.

$$\text{Payoff} = \frac{R_l + R_c + R_r}{3}, \tag{1}$$

where R_l, R_c, and R_r denote the ranks of the number of *likes*, *comments*, and *reposts*, respectively.

3. The topic payoff is calculated using formula (2) to quantify the overall influence of each topic and to construct the payoff matrix for NetLogo simulation. This matrix is used to simulate interactions between agents: when two agents meet, their payoffs are determined by the strategies (subtopics) chosen. For example, if agent 1 selects strategy 0 and agent 2 selects strategy 1, the payoff for agent 1 is the sum of the payoff of strategy 0 and strategy 1.

$$\text{Topic payoff} = \frac{\sum P_i}{N}, \tag{2}$$

where P_i is the payoff of each individual information piece under the topic, and N defines the total number of information pieces within the topic.

Through the aforementioned methods, this study quantifies the influence of information and assigns payoff to subtopics. Based on the payoff matrix, the effective integration of BERTopic and NetLogo is achieved: the influence of topics is embedded into multi-agent simulations, enabling agents to dynamically adjust their strategies and simulate the process of topic dissemination, thereby providing robust tool support for researching topic propagation mechanisms.

3. Experiments.

3.1. Experimental description. The dataset is sourced from the Weibo platform, covering the time range from October 6, 2024, to October 31, 2024, with approximately 20,000 Weibo posts. The data fields include username, content, publication time, number of reposts, number of comments, number of likes, and other relevant information.

In terms of model parameters, BERTopic uses the bert-base-chinese model, with UMAP for dimensionality reduction, HDBSCAN for clustering, and c-TF-IDF for topic representation. For NetLogo, the parameters include 310 agents, an initial strategy distribution of [249, 34, 27],

a payoff matrix of [[4.18, 3.61, 4.2], [3.61, 3.04, 3.63], [4.2, 3.63, 4.22]], a strategy revision probability of 0.06, and a noise parameter of 0.4.

3.2. Experimental results. Table 1 presents the classification results of the Weibo dataset using the BERTopic model, along with the Payoff for each subtopic. The dataset is divided into three subtopics: temperature changes, air pollution, and extreme weather. Over the 26-day period, subtopic 0 accounts for approximately 0.6 of the dataset, while subtopics 1 and 2 each constitute about 0.2, indicating significantly higher public attention to temperature changes. Subtopic 2 has the highest Payoff, suggesting that extreme weather events generate more frequent public interaction. Subtopic 0 follows with a slightly lower but still substantial Payoff, reflecting sustained public interest in daily temperature variations. In contrast, subtopic 1 has the lowest Payoff, indicating relatively weaker influence of related information.

Table 1. Subtopic classification and Payoff results

Subtopic	Keyword	Count	Payoff
0	temperature changes	12413	2.085529
1	air pollution	4192	1.518253
2	extreme weather	3953	2.110240

Then, to evaluate the performance of the NetLogo model, this study compares real and simulated topic dynamics. The real data (Figure 2) illustrates the composition of the three subtopics at 12-hour intervals, covering 52 time points in total. In the NetLogo simulation (Figure 3), each tick represents 12 hours in real time, so the comparison is made using the first 52 ticks of the simulation.

From the comparison results, it can be observed that the real data distribution and the simulated data distribution exhibit a high degree of consistency in their overall trends. Additionally, we attempted to adjust the parameters of the NetLogo model (setting the strategy revision probability to 0.12 while keeping the other parameters unchanged), which resulted in poorer experimental outcomes.

To assess the fit between simulated and real data, this study calculates the Root Mean Square Error (RMSE) and Normalized Root Mean Square Error (nRMSE) [5] between the real data and the simulated data with different parameters, thereby validating the effectiveness of the model and the rationality of the parameter configurations. The experimental results (Table 2) indicate that when the strategy revision probability (prob-revision) is set to 0.06, the RMSE value is 0.11 and the nRMSE value is 0.12, demonstrating a small deviation between the

simulated data and the real data, which validates the high accuracy of the NetLogo model. In contrast, when the prob-revision is set to 0.12, the RMSE value increases to 0.19 and the nRMSE value rises to 0.21, indicating a significant decline in model performance. This shows prob-revision's sensitivity and the need for tuning in the modeling process, providing critical insights for further model improvement.

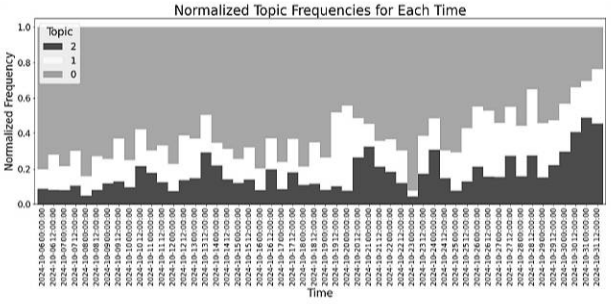


Figure 2. Original data distribution

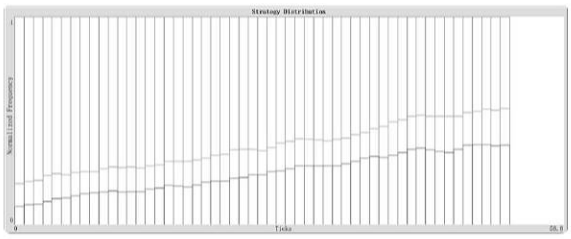


Figure 3. Simulate data distribution (The title of plot is strategy distribution, with the x-axis labeled as ticks and the y-axis as normalized frequency. In each tick, the three stacked areas from top to bottom represent the distribution of topic 0, topic 1, and topic 2, respectively.)

Table 2. Evaluation indicators

	RMSE	**nRMSE**
Method 1 (prob-revision=0.06)	0.11	0.12
Method 2 (prob-revision=0.12)	0.19	0.21

4. Conclusion. This study integrates BERTopic topic modeling and NetLogo multi-agent simulation based on evolutionary game approaches to propose a novel analytical framework for investigating the dissemination dynamics of different topics in social media. Experimental results demonstrate that BERTopic effectively extracts three subtopics from the Weibo dataset, while the NetLogo model successfully simulates the dynamic process of topic dissemination. Under appropriate parameter settings, the simulation results show a high degree of consistency with the real data, validating the model's high accuracy. Additionally, the experimental results highlight the importance of parameter optimization in the modeling process, providing critical insights for further model improvement.

References

1. Grootendorst M. BERTopic: Neural topic modeling with a class-based tf-idf procedure // arXiv preprint. 2022. arXiv:2203.05794.

2. Izquierdo L. R., Izquierdo S. S., Sandholm W. H. Agent-Based Evolutionary Game Dynamics. University of Wisconsin-Madison, 2019. 469 p.

3. Blei D. M., Ng A. Y., Jordan M. I. Latent Dirichlet Allocation // Journal of Machine Learning Research. 2003. Vol. 3. No Jan. P. 993–1022.

4. Marica D. Simulating an evolutionary multi-agent based model of the stock market // EcoForum (tefan cel Mare Univ. of Suceava, Romania, Faculty of Economics and Public Administration). 2015. Vol. 4. P. 1–42.

5. Yang J. M., Yang J. Y., Liu S., Hoogenboom G. An evaluation of the statistical methods for testing the performance of crop models with observed data // Agricultural Systems. 2014. Vol. 127. P. 81–89.

UDC 004

Feng Y.

Obtaining estimates for the rate of convergence of the gradient descent via machine learning

Recommended for publication by professor Abbasov M. E.

1. Introduction. In recent years, with the improvement of computing power and the development of big data, machine learning has received great attention in scientific research and industrial applications [1–4]. In this context, the convergence rate analysis of optimization algorithms has become a core issue in improving model training efficiency and resource utilization. The convergence rate reflects the number of iterations or time cost required for the algorithm to approach the optimal solution from the initial state. If we can estimate the convergence rate of the gradient descent algorithm, we can better understand the effect that the algorithm can achieve in a given time and avoid wasting a lot of computing resources without any improvement. However, traditional convergence rate estimation methods based on theoretical derivation face significant limitations in practical applications due to computational complexity when dealing with complex, nonlinear or high-dimensional optimization problems. In response to these challenges, we hope to replace these theoretical results with specially trained models.

2. Data generation process. We define a quadratic function $f(x_1, x_2) = \frac{1}{2}(ax_1^2 + bx_2^2)$ where $x = (x_1, x_2)$ is a two-dimensional vector, and a and b are randomly generated positive constants. The maximum eigenvalue of the Hessian matrix of the function at the minimum point is denoted by L, and the minimum is denoted by l. For this function, the Hessian matrix is a diagonal matrix with eigenvalues a and b. That is, $L = \max(a, b)$ and $l = \min(a, b)$. We randomly generate the parameters a and b, the initial coordinates $x_0 = \left(x_0^1, x_0^2\right)$ within the range [0, 0.5], and the step size h that satisfies $h = \frac{1}{2L}$ where L is the maximum eigenvalue of the current Hessian matrix.

Then use the gradient descent algorithm to optimize the objective function $f(x_1, x_2)$ for 1000 iterations. In each iteration, calculate the

Feng Yujia – graduate student, Saint Petersburg State University; e-mail: fx2000fyj@gmail.com, phone: +7(960)248-71-66

gradient and update the coordinates based on the step size h. Additionally, we calculate 1.1 multiples the Euclidean distance between the current coordinate and the function minimum point, and the relevant information (current number of iterations, current Euclidean distance, current step size, current eigenvalues L and l, and the initial coordinates) is recorded.
When the above optimization process is completed, we regenerate new values for a and b using the below 4 settings:

- Setting 1: a is in the range of (0, 10], b is in the range of (100, 400];
- Setting 2: a is in the range of (0, 400], $b = a$;
- Setting 3: a and b are both in the range of (0, 5];
- Setting 4: a and b are both in the range of (0, 500].

And regenerate new initial coordinates x_0 within the range [0, 0.5], as well as a new step size h that satisfies $h = \frac{1}{2L}$, and for each setting perform the optimization process 1000, 1000, 1000, 3000 times respectively. So this entire optimization process is repeated 6000 times, with each run consisting of 1000 iterations of gradient descent. Therefore, the total amount of data generated is 6,000,000 points. These data will record in detail the various parameters in each gradient descent process, including the number of iterations, Euclidean distance, step size, eigenvalues of the Hessian matrix and initial coordinate information, for subsequent analysis and research.

3. Model structure. In this paper, we use CatBoost model which is extremely efficient in processing large datasets, especially those with high-dimensional numerical data. The algorithm used by CatBoost is based on Gradient Boosted Trees (GBDT), which means that it improves the prediction performance of the model by gradually building trees one by one. The goal of each new tree is to minimize the residual of the previous tree. As shown in Figure 1 is the flow chart of our model.

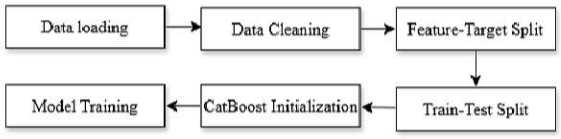


Figure 1. Flow chart

During the CatBoost model initialization process, we set the hyperparameters of the number of iterations to 7000, the learning rate to 0.02, the tree depth to 15, the loss function to Mean Absolute Error (MAE) which is less sensitive to outliersis, and the regularization parameter to 3. Because the dataset is large, we set the regularization parameter $\lambda = 3$ and tree depth 15 can effectively prevent overfitting. The total loss function after combining MAE and L2 regularization term with hyperparameter λ can be expressed as

$$\text{loss} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| + \lambda\sum_{i=1}^{k} w_j^2$$

and after we set λ equal to 3, ,our loss becomes

$$\text{loss} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| + 3\sum_{i=1}^{k} w_j^2,$$

where k is the number of leaf nodes in the tree, which is dynamic during training, w_j is the value predicted at this leaf node.

4. Model training. We first shuffle the dataset and split it into training, testing and validation sets with 60%, 20%, 20% of the data respectively. And use the above hyperparameters to train our model. The Figure 2 shows the loss value and evaluation index of the training set and the loss value of the test set.

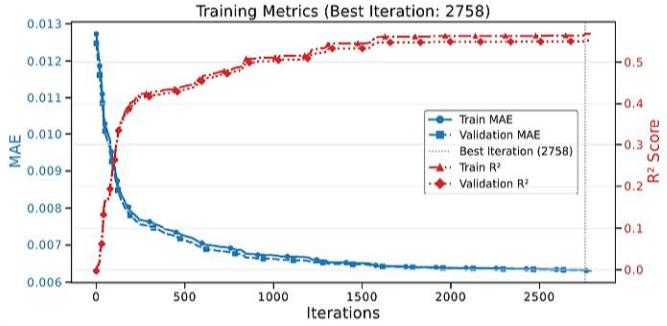


Figure 2. Learning curve

From the learning curve, we can see R^2 increases and MAE decreases which indicates that the model has good generalization ability. The training process reaches the best at 2758 iterations, and the curve does not change much after that, so we conclude that the model converges well and is well trained. Then we use metrics Mean Squared Error (MSE) and Mean Absolute Error (MAE) to evaluate the training and testing set. We find that for the training set MSE is 0.031 and MAE is 0.006, and for the testing set MSE is 0.034 and MAE is 0.006.

5. Model validation. To fully evaluate the performance of the CatBoost model, we use the four common optimization functions in two-dimensional space:

$$
\begin{aligned}
f(x_1, x_2) &= 0.26\left(x_1^2 + x_2^2\right) - 0.48 x_1 x_2, && \text{(Matyas)} \\
f(x_1, x_2) &= x_1^2 + x_2^2, && \text{(Sphere)} \\
f(x_1, x_2) &= 20 + \sum_{i=1}^{2}\left(x_i^2 - 10\cos(2\pi x_i)\right), && \text{(Rastrigin)} \\
f(x_1, x_2) &= 100\left(x_2 - x_1^2\right)^2 + x_1^2. && \text{(Rosenbrock)}
\end{aligned}
$$

The Data generation process is same as before. We now use the trained model to make predictions for the four optimization functions. We can plot the predicted values, actual values, and values of theoretical estimate for the convergence rate from [5] for the considered functions. These results are shown in Figure 3. And the formula of the theoretical estimate of k-th iteration is

$$
\text{upper boundary} = \frac{2 \cdot (f(x_0) - f^*)\|x_0 - x^*\|^2}{2\|x_0 - x^*\|^2 + k \cdot h(2 - Lh) \cdot (f(x_0) - f^*)},
$$

where x_0 represent the initial position, x^* represent the optimal (or lowest) point. The function values at these positions are $f(x_0)$ and f^* respectively. k denotes the current number of iterations, h is the step size, and L is the maximum eigenvalue of the Hessian matrix of the function at the minimum point. The Figure 3 shows the prediction, actual and book value of the four optimization functions.

From Figure 3, we find that starting from a certain high number of iterations, the predicted value of the function is close to the actual value and converges to the actual value asymptotically.

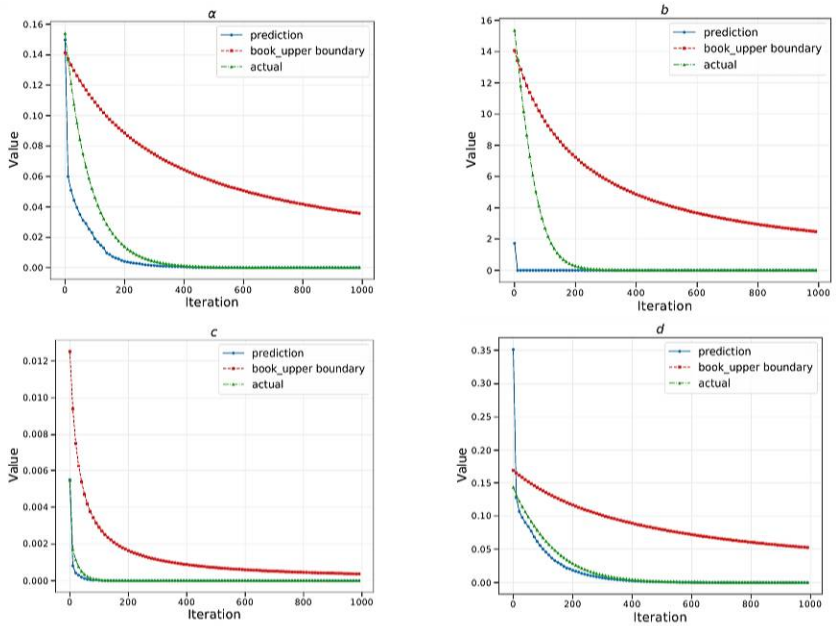


Figure 3. Prediction image of the optimization functions: (a) Sphere, (b) Rastrigin, (c) Matyas, (d) Rosenbrock

6. Conclusion. By constructing dataset, model selecting and training, and optimization function prediction visualization, the experimental results show that when the gradient descent step size h satisfies $h = \frac{1}{2L}$ where L is the maximum eigenvalue of the current Hessian matrix, the prediction effect of the model we trained on the optimization function meets our expectations. The performance on the optimization function verifies that this model is feasible, so our model can be used to predict the gradient descent convergence rate.

It may be possible to use the model to estimate the convergence rate of gradient descent for high-dimensional quadratic functions, which we will leave for future research.

References

1. Libbrecht M. W., Noble W. S. Machine learning applications in genetics and genomics // Nature Reviews Genetics. 2015. Vol. 16. No 6. P. 321–332.

2. Kourou K., Exarchos T. P., Exarchos K. P., et al. Machine learning applications in cancer prognosis and prediction // Computational and Structural Biotechnology Journal. 2015. Vol. 13. P. 8–17.

3. McCoy J. T., Auret L. Machine learning applications in minerals processing: A review // Minerals Engineering. 2019. Vol. 132. P. 95–109.

4. Kruppa J., Ziegler A., Knig I. R. Risk estimation and risk prediction using machine-learning methods // Human Genetics. 2012. Vol. 131. P. 1639–1654.

5. Nesterov Y. Introductory Lectures on Convex Optimization: A Basic Course. New York: Springer Science & Business Media, 2013. 236 p.

UDC 004.9

Gao T.

Structural and power-law degree distribution analysis of public transport networks in Chinese and Russian cities based on openstreetmap data and complex network theory

1. Introduction. Public transportation networks are fundamental infrastructures that shape urban development and social connectivity. Understanding their structural properties across different cultural and developmental contexts has become increasingly important for sustainable urban planning and policy making.

Complex network theory has emerged as a powerful framework for analyzing transportation systems. Barabsi and Albert's scale-free network model [1] established the theoretical foundation for understanding hub formation and network growth mechanisms. Watts and Strogatz's small-world theory [2] further enriched this framework by explaining the coexistence of high clustering and short path lengths in real-world networks. Building on these theoretical advances, Sienkiewicz et al. [3] demonstrated that public transport networks exhibit distinct topological patterns that can be quantitatively analyzed.

Recent methodological developments have created new opportunities for comprehensive network analysis. The availability of OpenStreetMap (OSM) data and specialized tools like Boeing et al.'s [4] OSMnx framework have enabled more sophisticated approaches to urban network analysis. Regional studies, such as Yang et al.'s [5] analysis of China's Bus Rapid Transit systems, have demonstrated the value of such approaches in understanding specific urban contexts.

Despite these advances, significant gaps remain in current research. Cross-cultural comparative studies of public transport networks are limited in scope and depth. There is insufficient understanding of how different urbanization stages influence network evolution. Furthermore, integrated methodological frameworks for analyzing multi-scale transport systems are largely lacking in the literature.

Gao T. – PhD student, Saint Petersburg State University; e-mail: st082013@student.spbu.ru, phone: +7(903)096-64-91

This study addresses these gaps by examining public transport networks in six major cities across China (Beijing, Shanghai) and Russia (Saint Petersburg, Moscow, Yekaterinburg, Kazan). These cities represent different stages of urban development and distinct planning traditions, providing an ideal setting for comparative analysis.

2. Research regions and data sources.

2.1. Regional overview. This study selects six representative cities: Beijing and Shanghai (China), and Saint-Petersburg, Moscow, Yekaterinburg, and Kazan (Russia). These cities exhibit distinct urban characteristics.

- Beijing: Chinas capital with polycentric development and heavy transit demand.
- Shanghai: A global economic hub with dense population and integrated transit-functional zone connections.
- Saint-Petersburg: A cultural city with geography-influenced layouts requiring balanced historical and modern transit solutions.
- Moscow: Russias capital and largest city with radial-circular transit systems.
- Yekaterinburg: An industrial center with transit systems evolving alongside industrial development.
- Kazan: A culturally diverse city requiring inclusive transit solutions.

Geography, population, urban layouts, and functional zones significantly influence transit configurations.

2.2. Data sources and preprocessing. Data derive from OpenStreetMap (OSM) and municipal planning documents. Bus stops, routes, and administrative boundary data were extracted from OSM, cleaned, and formatted using Python. New/old urban area classifications from planning documents were integrated with OSM data to create CSV/ JSON datasets containing administrative status, boundaries, and transit network information.

3. Public transport network modeling.

3.1. Urban boundary extraction. The city boundaries were obtained using the OSMnx `ox.geocode_to_gdf` function. Then, the data was simplified with the `simplify` function (where the tolerance parameter was set to 0.001) to preserve key geographic features while reducing

complexity. Based on these obtained boundaries, the bus stops within the urban limits were filtered.

3.2. Transit data parsing and initial processing. JSON files containing OSM transit elements (nodes, ways, relations) were processed as follows. *Nodes:* Valid stops (within city boundaries with "name" tags) were stored in `bus_stops` and `named_bus_stops`. *Ways:* Routes with less than 20 % missing nodes were retained in `bus_routes`, with stop-route associations recorded in `stop_routes`.

3.3. Relation data processing. OSM relations tagged as `type=route` with `route=bus/tram/trolleybus/share_taxi` were validated. Node members with roles `stop/platform/stop_position` were linked to routes, enhancing `stop_routes` completeness.

3.4. Administrative district integration. CSV files containing administrative regions (name, status, geometry) were parsed using Shapelys loads function to convert WKT geometries into spatial objects for district-level data filtering.

3.5. Network construction. It's made at two levels.

City-Level Networks: Nodes (bus stops) and directed edges (route connections) were built using `nx.DiGraph()`. Isolated nodes were removed. An example of the network for Saint-Petersburg with 6 468 stops and 34 032 routes is shown in Figure 1.

District-Level Networks: Nodes and edges within district boundaries were extracted, creating subnetworks. Isolated nodes were pruned, as shown in Figure 2.

For each administrative district, screening is carried out based on the existing bus stop and route data. Traverse the bus stops. According to whether the geographical coordinates are within the administrative district polygon, the qualified stops are stored in the `district_bus_stops` set, and those with names are stored in the `district_named_bus_stops` set. For bus routes, they are included in the district route set only when all nodes are in the identified bus stop set of this district. Subsequently, use the networkx library to create an administrative district bus network model. Add the screened stops as nodes to the graph, and use longitude and latitude to set the node position attribute pos. Sequentially set the adjacent nodes on the screened bus routes as directed edges to reflect the route direction and connection relationship. Finally, detect and delete isolated nodes (i.e., nodes with no connections) to ensure the model is valid and accurate.

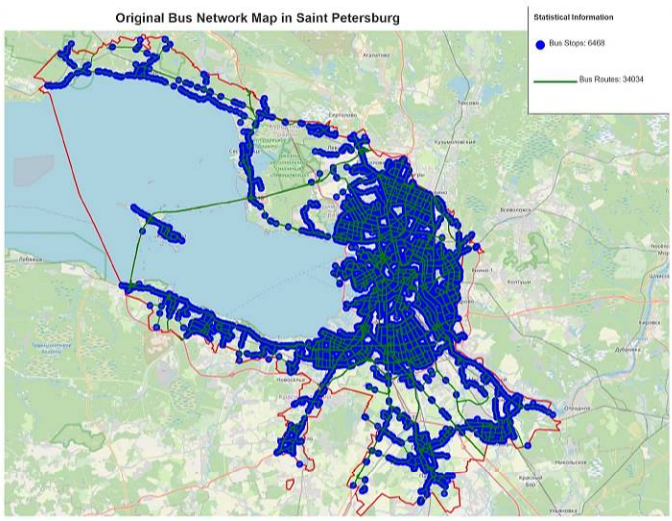


Figure 1. Bus network in Saint-Petersburg

4. Analysis of the public transport network structure. The analysis of the network structure of the entire city and each administrative district mainly focuses on the analysis of the out-degree and in-degree of the entire city, as well as the out-degree and in-degree of each administrative district. Scatter plots need to be drawn and function fitting is required. The main operation steps are as follows:

1. *Geographical scope definition and preprocessing*: Obtain the data of the city's boundary scope according to the city name and national information, which is used to screen the public transport elements within the city in the subsequent steps.

   Public transport network extraction: Parse the bus stops (nodes) and routes (paths) from the open map data, and screen out the valid stops (located within the city boundary or the administrative district boundary) and routes (with the proportion of missing nodes not exceeding the threshold).
2. N*etwork relationship supplementation*: Integrate the associated information of bus routes, improve the subordination relationship between stops and routes, and ensure that the routes passing through each stop are completely recorded.

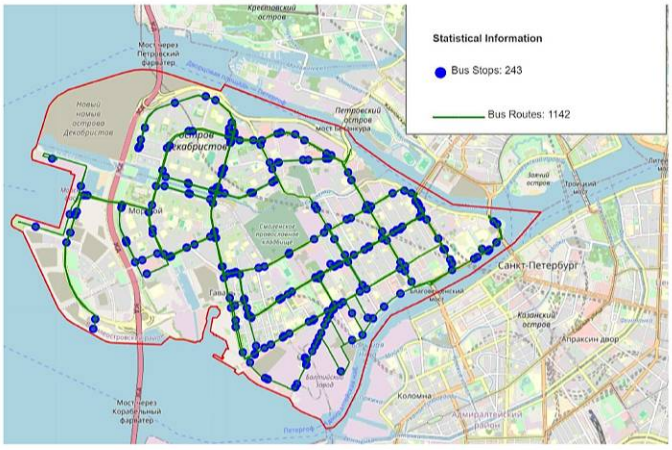


Figure 2. Bus network in Vasilyevsky Island

3. *Calculation of node out-degree and in-degree*: The out-degree of each stop is defined as the number of bus routes departing from the stop, and the in-degree is defined as the number of bus routes arriving at the stop.

   Statistics and probability calculation: Count the occurrence times of different out-degree and in-degree values among all stops, and calculate the corresponding probabilities of each degree (the occurrence times divided by the total number of stops).

4. *Data cleaning*: Exclude the stops with a degree of 0 (isolated nodes without route connections), and retain the valid data for fitting.

   Selection of power-law function fitting model: Use the power-law function $y = \frac{1}{x^k}$ to describe the degree distribution law, where x is the degree, y is the corresponding probability, and k is the power-law exponent [6].

5. *Parameter estimation*: Adjust k through an optimization algorithm to minimize the error between the function curve and the measured data.

6. *Fitting range control*: Perform the fitting only when the number of valid data points meets the condition (at least 2 points) to avoid invalid operations.

7. *Fitting effect evaluation – calculation of the coefficient of determination*: Measure the degree of coincidence between the fitting curve and the actual data through the R^2 value. The closer the value is to 1, the better the fitting effect.
8. *Superimposition of the fitting curve*: Draw the fitting curve on the scatter plot, mark the power-law exponent and the R^2 value, and visually compare the fitting results.

Taking Saint-Petersburg as an example, the scatter plots of the out-degree and in-degree of the entire city and the function fitting results are shown in Figure 3.

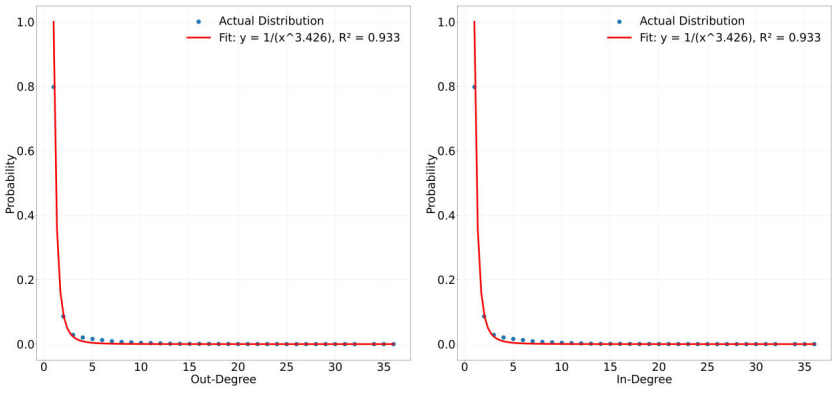


Figure 3. Fitting of out-degrees (left) and in-degrees (right) in Saint-Petersburg transit network

As shown in the figure, the k-value is 3.426, slightly larger than the range of [2, 3] for scale-free networks, and the R^2 value is 0.933, indicating a good function fitting result. Similarly, the scatter plots of the out-degree and in-degree of the administrative district and the function fitting results are shown in Figure 4 (taking the Vasilyevsky Island District of Saint-Petersburg as an example). It can be seen that the k-value is 3.231 and the R^2 value is 0.953, indicating a relatively good fitting effect.

Calculate the weighted average of the k-values of each administrative district (with the number of bus stops as the weight), and compare it with

the overall k-value of the city to verify whether the regional contributions are balanced. The indicators are as follows:

1. *Weight assignment*: The proportion of the number of bus stops in each administrative district to the total number of stops is used as the weight $w_i = \frac{N}{N_i}$.
2. *Weighted average formula:* Weighted average $k_{avg} = \sum_i (w_i \times k_i)$. For example, if the number of stops in a certain administrative district accounts for 20% of the total numbr $w_i = 0.2$and its $k_i = 1.5$, then the contribution of this district is 0.21.5=0.3.

   Comparison and verification of theoretical vs. actual value: Compare the calculated weighted average k_{avg} with the measured overall city value k.
3. *Difference analysis*: Calculate the absolute value of the difference between the two and determine whether it is within an acceptable range (for example, a difference <0.1 is considered consistent).

   Consistency test: If $k \approx k_{\text{avg}}$, it indicates that the overall characteristics of the city are equally contributed by each region, conforming to the hypothesis that "the whole is equal to the local average" [7]. If the difference is significant, it means that the network characteristics of certain administrative districts have a greater impact on the whole (for example, the core area with dense stops dominates the overall distribution).

Based on the above, taking Saint-Petersburg as an example, one can see, that the overall out-degree of the city is $k_{\text{out}} = 3.426$ and $R^2 = 0.933$; the overall in-degree of the city is $k_{\text{in}} = 3.426$ and $R^2 = 0.933$.

Verification of the mean-value relationship for the out-degree: the theoretical weighted average out-degree is $k_{\text{out}} = 3.692$, while the actual overall out-degree of the city is $k_{\text{out}} = 3.426$, with an absolute difference of 0.265.

Verification of the mean-value relationship for the in-degree: the theoretical weighted average in-degree is $k_{\text{in}} = 3.692$, while the actual overall in-degree of the city is $k_{\text{in}} = 3.426$, with an absolute difference of 0.265.

Comparison between new and old urban areas: the "express-stop" model (with high k-mean values and R^2 dominated by urban planning in new urban areas prioritizes efficiency, has a high hub concentration, but less stop coverage. The decentralized networks formed in old urban areas due to historical reasons (with low k-values) can meet the demands

of intensive travel (more stops), but overlapping routes may reduce efficiency.

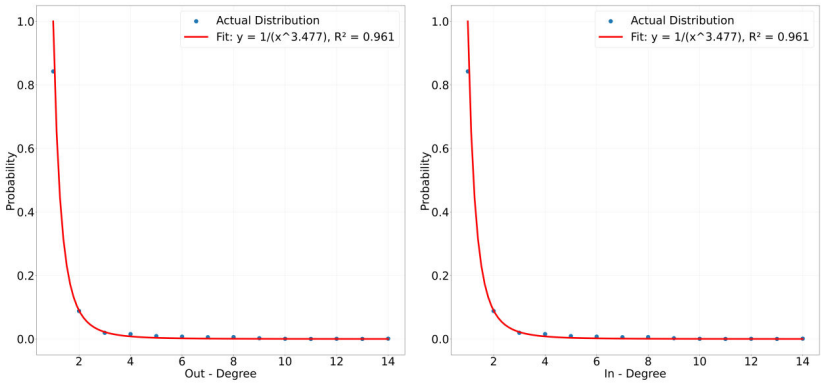


Figure 4. Fitting of out-degrees (left) and in-degrees (right) for Vasilyevsky Island

5. Conclusions and prospects. An analysis of the public transport networks in six Chinese and Russian cities reveals several key characteristics. All networks are scale-free with higher power-law exponent k values (3.4-3.8) than classical ones ($k \in [2, 3]$), indicating strong hub concentration. Cross-district edges dominate (e.g., 69.1% in Saint-Petersburg) over internal routes (30.9%). The city-level k value is 0.265 lower than the district average, as cross-district hub-agglomeration weakens local linear superposition. New urban areas ($k = 3.833$, $R^2 = 0.970$) follow a planned express-stop model with high efficiency but less coverage, while old areas ($k = 3.584$) can meet intensive travel needs but may suffer efficiency losses from overlapping routes.

References

1. Barabsi A.-L., Albert R. Emergence of scaling in random networks // Science. 1999. Vol. 286. Iss. 5439. P. 509–512.

2. Watts D. J., Strogatz S. H. Collective dynamics of small-world networks // Nature. 1998. Vol. 393. P. 440–442.

3. Sienkiewicz J., Hoyst J. A. Statistical analysis of 22 public transport networks in Poland // Physical Review E. 2005. Vol. 72. Iss. 4. Art. no 046127.

4. Boeing G. OSMnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks // Computers, Environment and Urban Systems. 2017. Vol. 65. P. 126–139.

5. Yang L., Chau K. W., Chu X. Accessibility-based premiums and proximity-induced discounts stemming from bus rapid transit in China // Sustainable Cities and Society. 2019. Vol. 48. Art. no 101561.

6. Faloutsos M., Faloutsos P., Faloutsos Ch. On power-law relationships of the Internet topology // ACM SIGCOMM Computer Communication Review. 1999. Vol. 29. Iss. 4. P. 251–262.

7. Hznagy A., Fi I., London A., Nemeth T. Complex network analysis of public transportation networks: A comprehensive study // 2015 International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS), Budapest, Hungary, 2015. P. 371–378.

8. Louf R., Barthelemy M. How congestion shapes cities: from mobility patterns to scaling // Scientific Reports. 2014. Vol. 4. Art. no 5561.

UDC 519.1

Gao Y., Xu H.

Reliability of communication network

Recommended for publication by professor Petrosian O. L.

1. Introduction. Let $G = (V, E)$ be a *graph (network)*, where V is a set of vertices (nodes) and $E \subset V \times V$ is a set of edges (links). Let Ω be a set of events: edges from E fail or don't fail over one unit of time (depending on the practical application, it could be one year/month/day/hour/...). We also call it a *functional/dysfunctional* edge. Each edge $e \in E$ has an attribute $\alpha_e : \Omega \to [0, 1]$, which is the probability that this edge remains functional during a unit of time, and $f_e := 1 - \alpha_e$ is the probability that e became dysfunctional during a unit of time, i.e. broken. All α_e are independent identically distributed random variables [1].

Definition 1 [2]. Let $v_1, \ldots, v_k$ be a sequence of vertices from V. A *path*, $v_1 \rightsquigarrow v_k$, in G is an event in Ω, such that each consecutive pair v_i, v_{i+1} from that sequence has a functional edge in E. A path, where its sequence of vertices doesn't have repetitions, is called *simple path.* A simple path is called a *chain* if the degree of its internal nodes equal two.

Definition 2. If for each pair of vertices $u, v \in V$ in graph $G = (V, E)$ there exists a path $v \rightsquigarrow u$, we say G is *connected.* Otherwise, G is *not connected (disconnected, failed).*

Note that connected or disconnected graph is an event in Ω, hence, it has a probability. There are many ways to understand reliability of network depending on the specific features one wants to capture. Here we use the following

Definition 3 [3]. The probability that $G = (V, E)$ is connected is *reliability* of graph G, and denoted as $R(G)$. Denote $F(G) := 1 - R(G)$ to be the probability that G is not connected.

To avoid boring the reader, we skip standard definition of *tree graph* and ring graph by presenting Figure 1 of examples.

Gao Yang – graduate student, Saint Petersburg State University; e-mail: st115833@student.spbu.ru, phone: +7(906)226-89-96

Xu Huimin – graduate student, Saint Petersburg State University; e-mail: xuhuimin234@gmail.com, phone: +7(906)270-19-30

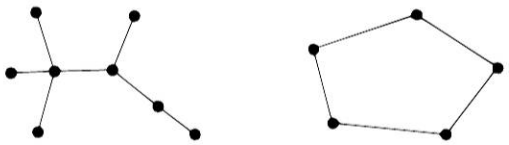


Figure 1. Tree and ring

2. Goal. The main focus of this work is to find practical ways to calculate reliability of graphs, including exact formulas for graphs of some topology and approximate algorithms for general topology graphs.

The presented Definition 3 of reliability brings an obvious fact that a disconnection of a single node leads to a failure of the whole graph. For large practical networks (telecommunication, road map etc.), such event happens all the time, which makes this reliability meaningless for applications. Thus, we recommend considering this work in a view of relatively small graphs. Perhaps, data storage centers and the core of telecommunication network.

3. Necessary results. Here we introduce the definitions and statements on special graphs we need for the main result presentation.

Theorem 1. *Let $G = (V, E)$ be a ring graph, then*

$$R(G) = \prod_{e \in E} \alpha_e \left(1 + \sum_{e \in E} \frac{f_e}{\alpha_e} \right). \tag{1}$$

Proof. Since G is a simple cycle, there are only two events, when G is a connected graph: (i) Every edge is functional, $R_1 = \prod_{e \in E} \alpha_e$; (ii) Only one edge has failed

$$R_2 = \sum_{e_i \in E} (f_{e_i} \prod_{e_j \neq e_i} \alpha_{e_j}) = \sum_{e_i \in E} (f_{e_i} \prod_{e_j \neq e_i} \alpha_{e_j} \frac{\alpha_e}{\alpha_e}) = \prod_{e \in E} \alpha_e \sum_{e \in E} \frac{f_e}{\alpha_e}.$$

Now summing up R_1 and R_2, we obtain the needed formula for $R(G)$.

Note 1. If the attributes of each edge in G are the same, i.e. $\alpha_e = \alpha, f_e = f, \forall e \in E$, then $R(G) = \alpha^{|E|} + |E|\, f \alpha^{|E|-1}$.

For convenience of graph construction, we describe the procedure of *subgraph contraction*: Let $G = (V, E)$ be a graph and $H = (N, L)$ be an induced subgraph of G, meaning $N \subset V$ and no edges from $E \setminus L$ have

vertices from N on both ends. Denote E_N to be a subset of E connecting $V \setminus N$ and N. We say the graph $G' = \big(\{V \setminus N\} \cup a, \{E \setminus L\} \cup E_a\big)$, where a is a new vertex to substitute graph H and E_a is the set of edges E_N relinked to a instead of N on one end, is a result of contraction of subgraph H in G.

Definition 4. Graph $G = (V, E)$ is called *generalized tree/ring*, if it is possible to conduct a sequence of ring subgraph contractions of G to obtain a tree/ring graph in such a way that each new vertices obtained by a subgraph contraction does not get involved in other subgraph contractions. The example is shown in Figure 2.

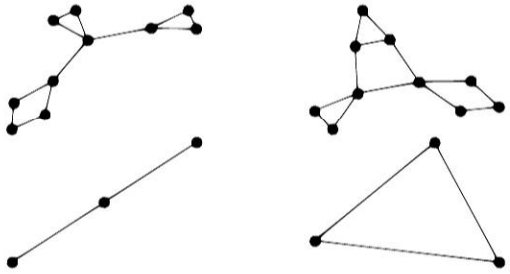


Figure 2. Generalized tree and generalized ring

Theorem 2. *Let $G = (V, E)$ be a generalized tree with k rings $C_1, C_2, \ldots, C_k$, then*

$$R(G) = \prod_{e \in E} \alpha_e \cdot \prod_{t=1}^{k} \left(1 + \sum_{e \in E_t} \frac{f_e}{\alpha_e}\right), \tag{2}$$

E_t is the edge set of ring $C_t, t = 1, 2, \ldots, k$.

Proof. When $k = 1$, it means that G is a single ring. We have proved it. Now assume that the theorem holds when G has k cycles. Now consider changing G into $G' = (V', E')$ by adding a ring C_{k+1}, at this time the number of rings is $k + 1$ and $E' - E = E_{k+1}$. When calculating the reliability of G', the connectivity of G' must be under the premise that G is connected, and the newly added rings C_{k+1} must

also be connected in time. So we have

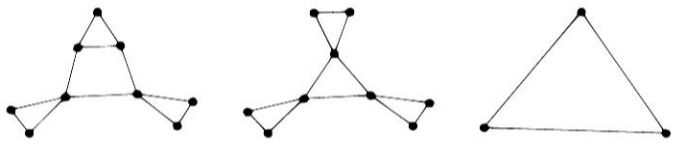
$$\begin{aligned} R(G') &= R(G) \cdot R(C_{k+1}) \\ &= \prod_{e \in E} \alpha_e \times \prod_{t=1}^{k} \left(1 + \sum_{e \in E_t} \frac{f_e}{\alpha_e}\right) \times \prod_{e \in E_{k+1}} \alpha_e \left(1 + \sum_{e \in E_{k+1}} \frac{f_e}{\alpha_e}\right) \\ &= \prod_{e \in E'} \alpha_e \times \prod_{t=1}^{k+1} \left(1 + \sum_{e \in E_t} \frac{f_e}{\alpha_e}\right). \end{aligned} \tag{3}$$

From (2) we can see that the reliability of a generalized tree is the product of the reliability of the cycle and the reliability of the remaining edges.

Corollary 1. *The location of the connection nodes between different rings in generalized tree doesn't matter.*

Corollary 2. *When we calculate the reliability of a generalized tree, we can contract a ring C to a node x with $R(x) = R(C)$.*

Note 2. Later, when we calculate the reliability of a generalized ring, the location of the connection nodes between different rings will have an impact on the results and we can't contract a ring C to a node x with $R(x) = R(C)$.



(a) G_1 (b) G_2 (c) G_3

Figure 3. Generalized ring

Example 1. In Figure 3, G_1 and G_2 are both generalized rings containing three rings, which are graphs composed of different positions of connection nodes between rings. The attributes of each edge in G_1 and G_2 is 0.9. A Monte-Carlo simulation (3000 times) of these events gives $R(G_1) = 0.905$, $R(G_2) = 0.892$. And we can contract three rings in G_2 to become G_3 with any node in G_3 has a reliability of 0.972.

Definition 5. For a connected graph G , a subset E' of the edge set $E(G)$ is an *edge cut* of G if $G - E'$ is disconnected; E' is an *k-edge cut* if $|E'| = k$ and $\forall E'' \subset E'$ such that $E'' \neq E'$ implies $G - E''$ is still

connected. An edge $e \in E$ of G is an *cut edge* of G if E is an k-edge cut of G for some k.

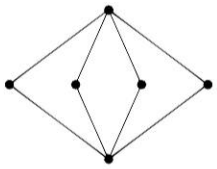


Figure 4. Multi-chain

Theorem 3. *Let σ be a chain in some graph G, where there are at least two paths between the two endpoints of the chain σ. Then its failure probability is*

$$\begin{aligned} F(\sigma) &= F\left(\bigvee_{1\le i<j\le n}(e_i \wedge e_j)\right) \\ &= \sum_{k=2}^{n}\left((-1)^k\cdot(k-1)\sum_{1\le i_1<i_2<\cdots<i_k\le n}F(e_{i_1}\wedge\cdots\wedge e_{i_k})\right), \end{aligned} \tag{4}$$

n is the length of chain σ, $n \ge 2$.

Proof. This follows from the formula of ring availability Theorem 1.

Definition 6. It is a *multi-chain graph* if there are only two points in this graph with degree more than or equal to 3, and the degree of the other nodes is 2 (see Figure 4).

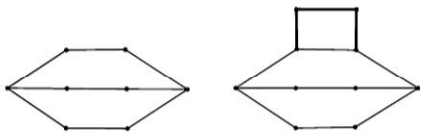


Figure 5. Adding chain to chains in multi-chain

Theorem 4. *Let H_1 be a multi-chain graph with m chains $\sigma_1, \sigma_2, \ldots, \sigma_m$ connected by nodes s and t, then*

$$F(H_1) = \prod_{e\in H_1}\alpha_e\sum_{e_i\in\sigma_i}\frac{\prod_{e_i}f_{e_i}}{\prod_{e_i}\alpha_{e_i}} - \prod_{i=1}^{m}R(\sigma_i) + 1. \tag{5}$$

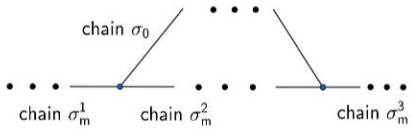


Figure 6. Adding chain σ_0 to chain σ_m

Proof. The reasons for H_1 to be disconnected are: 1) at least one chain $j \in \{1, ..., m\}$ is broken (each chain must have 2 or more edge cut), or 2) each chain has 1-edge cut. 1) and 2) can be seen as independent events conditional on the 1-edge cut of i-th chain. So, this means $F(H_1) = F(\sigma_1 \text{ broken} \vee \cdots \vee \sigma_m \text{ broken}) + F(m\text{-edge cut})$.

Here, m-edge cut means that each chain breaks one edge. In that way, assume Γ be the set of edge cuts, $\gamma = \{\gamma_1, \gamma_2, \ldots, \gamma_m\} \in \Gamma$, $\gamma_i \in \sigma_i$, so $|\Gamma| = \prod_{i=1}^{m} |\sigma_i|$, we get

$$F(m\text{-edge cut}) = \sum_{\gamma \in \Gamma} \prod_{i=1}^{m} f_{\gamma_i} \cdot \prod_{j=1, j \neq \gamma_i}^{|\sigma_i|} \alpha_j = \prod_{j \in H_1} \alpha_j \sum_{\gamma \in \Gamma} \frac{\prod_{i=1}^{m} f_{\gamma_i}}{\prod_{i=1}^{m} \alpha_{\gamma_i}}. \quad (6)$$

Notice that

$$\begin{aligned} &F(\sigma_1 \text{ broken} \vee \sigma_2 \text{ broken} \vee \cdots \vee \sigma_m \text{ broken}) \\ &\quad = 1 - R(\sigma_1)R(\sigma_2)\cdots R(\sigma_m) = 1 - \prod_{i=1}^{m} R(\sigma_i). \end{aligned} \quad (7)$$

Now to get closer to communication network, we consider the case of adding chain to chains in multi-chain graph, as shown in Figure 5.

Let H_1 be a multi-chain graph with m ($m \geq 3$) chains $\sigma_1, \sigma_2, \ldots, \sigma_m$ and H_2 be a graph formed by connecting a new chain to some intermediate nodes of some chain of graph H_1. Without loss of generality, assume H_2 is the graph formed by adding chain σ_0 to chain σ_m of H_1, then H_2 has $m + 3$ chains and the original chain σ_m in the H_1 has become 3 chains $\sigma_m^1, \sigma_m^2, \sigma_m^3$ (link up in sequence) as Figure 6.

Consider all disconnected cases of H_2: 1) the chain $\sigma_i \subset H_2$ is broken (at least two edges in the chain are broken); 2) chain σ_m^1 and σ_m^3 both have one edge broken; 3) chain σ_0, σ_m^1 and σ_m^2 each have one edge broken or chain σ_0, σ_m^3 and σ_m^2 each have one edge broken; 4) chain $\sigma_1, \sigma_2, \ldots, \sigma_{m-1}$ and σ_m^1 each have one edge broken or

chain $\sigma_1, \sigma_2, \ldots, \sigma_{m-1}$ and σ_m^3 each have one edge broken; or 5) chain $\sigma_1, \sigma_2, \ldots, \sigma_{m-1}$ and σ_0, σ_m^2 each have one edge broken. These are 5 independent events. Therefore, the failure probability of H_2 can be approximated by the sum of the probabilities of the 5 events which is $F(H_2) \approx 1 - \prod_{\sigma_i \subset H_2} R(\sigma_i) + F_0$, where F_0 is given by

$$F_0 = \prod_{e \in E} \alpha_e \left(\sum_{\substack{e_m^1 \in \sigma_m^1 \\ e_m^3 \in \sigma_m^3}} \frac{f_{e_m^1} f_{e_m^3}}{\alpha_{e_m^1} \alpha_{e_m^3}} + \sum_{\substack{e_m^1 \in \sigma_m^1 \\ e_m^2 \in \sigma_m^2 \\ e_0 \in \sigma_0}} \frac{f_{e_m^1} f_{e_m^2} f_{e_0}}{\alpha_{e_m^1} \alpha_{e_m^2} \alpha_{e_0}} + \sum_{\substack{e_m^3 \in \sigma_m^3 \\ e_m^2 \in \sigma_m^2 \\ e_0 \in \sigma_0}} \frac{f_{e_m^3} f_{e_m^2} f_{e_0}}{\alpha_{e_m^3} \alpha_{e_m^2} \alpha_{e_0}} + \right. \tag{8}$$

$$\left. \sum_{\substack{e_m^1 \in \sigma_m^1 \\ e_i \in \sigma_i \\ i=1,\ldots,m-1}} \frac{f_{e_m^1} \prod f_{e_i}}{\alpha_{e_m^1} \prod \alpha_{e_i}} + \sum_{\substack{e_m^3 \in \sigma_m^3 \\ e_i \in \sigma_i \\ i=1,\ldots,m-1}} \frac{f_{e_m^3} \prod f_{e_i}}{\alpha_{e_m^3} \prod \alpha_{e_i}} + \sum_{\substack{e_m^2 \in \sigma_m^2 \\ e_i \in \sigma_i \\ i=0,\ldots,m-1}} \frac{f_{e_m^2} \prod f_{e_i}}{\alpha_{e_m^2} \prod \alpha_{e_i}} \right).$$

4. k-edge cut search algorithm. In this section, we investigate to find k-edge cuts by searching k-flow using Ford–Fulkerson algorithm [4]. Let $\mathcal{P} = \{P_1, P_2, \ldots, P_k\}$ denote the set of k disjoint paths between *source* and *sink*.

To express all k-edge cuts of this graph, define the Cartesian product of the edges of each path in $\mathcal{P}$ as $\Lambda = \prod_{i=1}^{k} E(P_i)$. Each element of Λ is a tuple $(e_1, e_2, \ldots, e_k)$ where $e_i \in E(P_i)$ and is also an event of graph failure.

The following phases of the algorithm 1 dominate in its complexity: 1) Node pair iteration: $O(|V|^2)$ for all node pairs; 2) Path generation: $O(|V|^2 \cdot k \cdot |E|)$ to find k disjoint paths via Ford–Fulkerson; 3) Combinatorial verification: $O(|V|^2 \cdot m^k \cdot (|V| + |E|))$ to test m^k edge combinations for connectivity, where m is the maximal path length. Therefore, the overall complexity is $O(|V|^2 \cdot (k \cdot |E| + m^k \cdot (|V| + |E|))$.

Furthermore, all edge cuts $\mathcal{C}_1, \mathcal{C}_2, ..., \mathcal{C}_l$ whose size is less than or equal to the maximal degree of network can be obtained if k can be identified automatically. We can implement *networkx.edge_connectivity(G, s=source, t=sink)* to get k. With the maximal degree increasing, we can evaluate network failure probability by $F(G) \approx P(\bigvee_{i=1}^{l} \mathcal{C}_i \text{ fail})$ with probability of inclusion-exclusion principle and call it Algorithm 1.

Algorithm 1: k-edge cut search algorithm

Input: Given a network $G = (V, E)$, the edge cut size k
Output: All k-edge cuts in the network
$\Lambda \leftarrow [\,]$, $\Gamma \leftarrow [\,]$;
for any pair of nodes $u, v \in V(G)$ $(u \neq v)$ **do**
  Get k disjoint paths $P_1, P_2, \ldots, P_k$ by Ford–Fulkerson algorithm;
  $\Lambda \leftarrow$ Cartesian product $\prod_{i=1}^{k} E(P_i)$;
end for
for each $(e_1, e_2, \ldots, e_k) \in \Lambda$ **do**
  Delete edges $e_1, e_2, \ldots, e_k$;
  if G is not connected **then**
    Add $(e_1, e_2, \ldots, e_k)$ to Γ;
  end if
  Add edges $e_1, e_2, \ldots, e_k$ back to the network;
end for
return Γ

5. Simulation results. In this section, 40 connected networks with 8, 9 and 10 nodes are generated randomly. The link reliability is within the range $[0.8, 0.99]$ and the probability of link creation range $[0.3, 0.6]$. The number of edge cuts in dataset ranges from 7 to 21. We compare the results between Algorithm 1 and Monte-Carlo simulation. Observe that Algorithm 1 outperforms Monte-Carlo simulation when the number of edge cuts is relatively small as Figure 7 shows. In Figure 7, the figure above shows the difference in network failure probability between the Algorithm 1 and the Monte-Carlo simulation, and the figure below shows the changes in $T(simulation) - T(algorithm)$ for different graphs where $T(simulation)$ and $T(algorithm)$ represents the running time of Monte-Carlo simulation and Algorithm 1.

6. Conclusion. Communication network reliability topic is studied. This paper derives the reliability metrics for rings, generalized trees with contractibility insights. Additionally, we have explored the infeasibility of chain contraction within the context of multi-chain graphs. This paper not only establishes a theoretical formula, but also explores the estimation algorithm. Building upon classical maximum flow algorithms, we

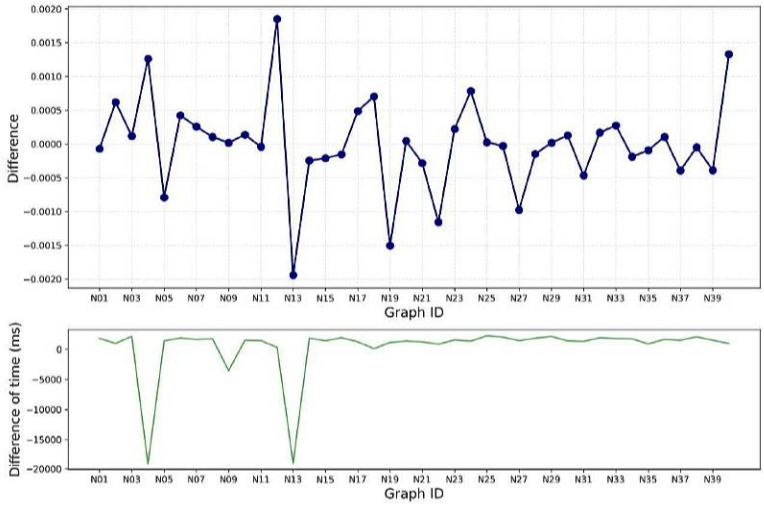


Figure 7. Results comparison

have applied k-edge cut search algorithm, leveraging the k-flow path approach, to estimate the reliability of graphs.

Communication network reliability analysis is complex challenge. Although connectivity-based methods provide a topological graph perspective that incorporates probabilistic considerations, these methods are limited to the number of nodes and topology. In the future, reliability research of networks with more complicated topology might be interesting.

References

1. Ball M. O., Colbourn C. J., Provan J. S. Network reliability // Handbooks in operations research and management science. 1995. Vol. 7. P. 673–762.

2. West D. B. Introduction to graph theory. Upper Saddle River: Prentice Hall, 2001. P. 5–11.

3. Satyanarayana A., Chang M. K. Network reliability and the factoring theorem // Networks. 1983. Vol. 13. No 1. P. 107–120.

4. Ford L. R. Jr., Fulkerson D. R. Maximal flow through a network // Canadian Journal of Mathematics. 1956. Vol. 8. P. 399–404.

UDC 004.912

Li J., Blekanov I. S.

Text encoding approaches for topic modeling in the social media platform

1. Introduction. In the digital age, the exponential growth of unstructured text data from social media, news, academic papers, and user comments has created both opportunities and challenges for information extraction and knowledge discovery. [1] Traditional data mining techniques, while effective for structured numerical data, struggle with the complexity, ambiguity, and dynamic nature of natural language. To address these challenges, topic models have emerged as a powerful tool for uncovering latent topic structures in text, enabling applications such as trend analysis, recommendation systems, and decision support. These models are particularly valuable in social network content analysis, where they help reveal user behavior, information dissemination patterns, and social dynamics by identifying implicit topic distributions.

We conducted a comparative analysis of various neural network encoding models based on the Transformer architecture [2], integrating them with clustering methods for effective topic detection. [3] Our experiments focused on a dataset of COVID-related discussions from Twitter, collected between November 24, 2021, and January 23, 2022. The methodology pipeline includes text encoding, dimensionality reduction, and clustering, with detailed steps and results discussed in subsequent sections. The findings highlight the model's effectiveness in topic detection, offering insights for future research and practical applications in text analysis and knowledge discovery.

2. Methods and materials. The model employed in this study is based on an approach similar to hierarchical topic modeling, which facilitates topic detection through three core steps: semantic embedding, dimensionality reduction, and clustering, as illustrated in Figure 1.

Li Jing – PhD student, Saint Petersburg State University; e-mail: st082014@student.spbu.ru, phone: +7(929)100-80-22

Blekanov Ivan Stanislavovich – associate professor, Saint Petersburg State University; e-mail: i.blekanov@gmail.com, phone: +7(812)363-65-37

This research has been supported by the project "Cumulative public opinion and deliberation: Theorizing, research methodologies, and quality assessment of today's deliberative process", ID 128786104, 2025

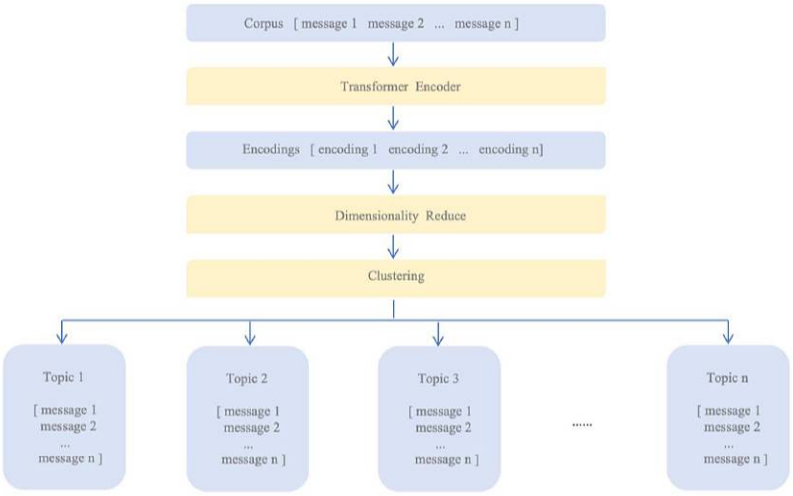


Figure 1. BERTopic Structured Model

2.1. Semantic encoding.

- **MPNet.** The MPNet model [4] combines Masked Language Modeling (MLM) and Permuted Language Modelling (PLM) to overcome BERT and XLNet limitations, capturing token dependencies and aligning pre-training with fine-tuning. Its unique attention mask, dual-stream self-attention, and positional encoding boost performance. We used the all-mpnet-base-v2 version.

- **MiniLM.** The MiniLM model [5], a distilled variant of BERT, reduces parameters for smaller size and faster inference while retaining Transformer encoder functionality. It uses distillation objectives to optimize sentence-level embeddings. We employed the all-MiniLM-L6-v2 and all-MiniLM-L12-v2 versions.

- **E5.** The E5 model [6] generates high-quality text embeddings via weakly supervised contrastive pre-training, using large-scale text-to-data sets like CCPAirs for contrastive learning. It differentiates positive and negative samples and fine-tunes with limited labeled data. We used the e5-small-v2 variant.

- **GTE.** The GTE model [7] is a generalized text embedding framework that utilizes multi-stage contrastive learning. The central concept involves employing contrastive learning to derive generalized embeddings from extensive collections of unsupervised text pairs, followed by supervised fine-tuning to enhance performance on specific tasks. We utilized the gte-small variant of this model.

2.2. Dimensionality reduction processing. To reduce computational complexity while preserving key semantic features, we employed the UMAP algorithm [8] to map the high-dimensional vector space to a low-dimensional representation. Through hyperparameter optimization, the dimensionality was ultimately reduced to 50, effectively retaining the topological characteristics of the original vector space.

2.3. Topic clustering. We utilized the HDBSCAN [9] algorithm for density clustering, which automatically determines the optimal number of clusters by constructing a hierarchical clustering tree. This algorithm excels at processing noisy data, making it particularly well-suited for the analysis of highly sparse datasets.

3. Experiment. In this study, we comparatively analyze the impact of text embedding models on topic detection and conduct a series of experiments. The experimental process begins with the conversion of text data into embeddings using text encoding models. This is followed by dimensionality reduction of the embeddings, after which clustering analysis is performed on the reduced data to identify distinct topics. The subsequent sections provide a detailed description of the experimental dataset, evaluation metrics, and results.

3.1. Dataset. We collected over 100,000 tweets concerning users' discussions about COVID-19 from Twitter between November 24, 2021, and January 23, 2022 using the Twitter API. The collected data was pre-processed and cleaned to ensure its quality and relevance for further analysis.

3.2. Parameter setting. We employed the UMAP dimensionality reduction method, setting the target to 50 dimensions for subsequent clustering tasks. For topic clustering, we utilized the HDBSCAN clustering algorithm, with parameter settings that include minimum sample size and distance metric. Specifically, we set the minimum cluster size to 1,500 and used the default Euclidean distance as the distance metric to optimize clustering effectiveness.

3.3. Evaluation indicators. In this thesis, we utilize five evalu-

ation metrics to comprehensively assess the effectiveness of topic detection, namely Silhouette Coefficient [10], Calinski-Harabasz Index [11], Davies-Bouldin Index [12], UMass [13], and NPMI [14]. Through the collective evaluation of these metrics, we effectively measure clustering cohesion, separation, and thematic consistency, thereby providing a more accurate analysis of theme detection quality.

3.4. Results. The experimental results, as illustrated in Table 1, indicate that various text embedding models significantly influence topic detection when employing the UMAP dimensionality reduction method and HDBSCAN clustering algorithm. Among them, combined with the gte-small model demonstrates the best performance.

Table 1. Topic detection results with different text encoding models assessed by various evaluation metrics

Method	Silhouette Coefficient	Calinski Harabasz	Davies Bouldin	UMass	NPMI
all-mpnet-base-v2	−0.00511	**621.695**	**5.235**	−2.883	0.053
all-MiniLM-L6-v2	−0.00485	589.023	5.363	−3.126	0.06
all-MiniLM-L12-v2	−0.00519	**597.737**	5.394	−2.632	0.071
e5-small-v2	−0.00856	354.857	6.79	−3.08	**0.082**
gte-small	−0.00025	596.122	**5.295**	−2.728	**0.112**

Moreover, as illustrated in Figure 2, the application of the gte-small text encoding model for topic detection in user discussions resulted in the identification of 20 distinct discussion topics related to COVID.

4. Conclusion. In this paper, we conduct a comparative analysis of various text encoding models for topic detection, leveraging the BERTopic framework's flexibility by decoupling document embedding and topic representation processes, enabling effective use of diverse preprocessors. However, the model's assumption that each document addresses only one topic and its potential redundancy in topic interpretation due to word correlations present limitations. Future research will

focus on exploring topic branches and subtopics, abstracting and summarizing content to better observe topic evolution and improve analytical accuracy, while enhancing the model's flexibility to adapt to diverse data types and application scenarios for better practical performance.

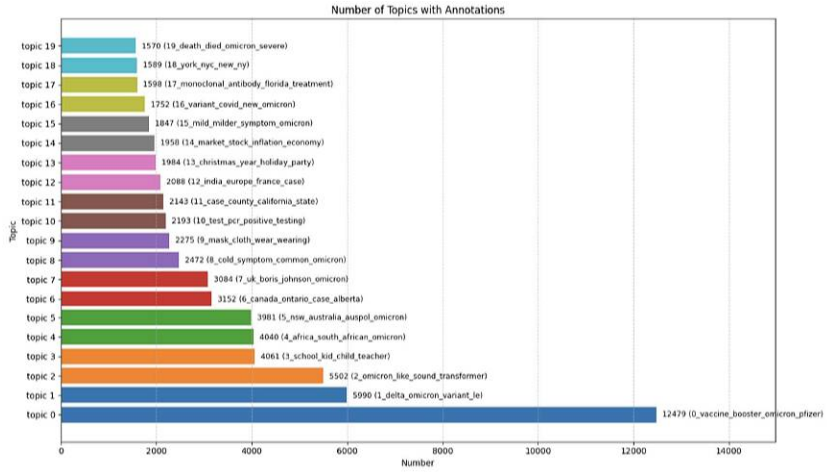


Figure 2. The number of annotated topics detected with the gte-small model

References

1. Lu X. S. et al. Clustering-algorithm-based rare-event evolution analysis via social media data // IEEE Transactions on Computational Social Systems. 2019. Vol. 6. No 2. P. 301–310.

2. Vaswani A. et al. Attention is all you need // NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems. 2017. P. 6000–6010.

3. Grootendorst M. BERTopic: Neural topic modeling with a class-based TF-IDF procedure // arXiv preprint, 2022. https://arxiv.org/abs/2203.05794v1

4. Song K. et al. MPNet: Masked and permuted pre-training for language understanding // NIPS'20: Proceedings of the 34th Interna-

tional Conference on Neural Information Processing Systems. 2020. Art. no. 1414. P. 16857–16867.

5. Wang W. et al. MINILM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers // NIPS'20: Proceedings of the 34th International Conference on Neural Information Processing Systems. 2020. Art. no. 485. P. 5776–5788.

6. Wang L. et al. Text embeddings by weakly-supervised contrastive pre-training // arXiv preprint, 2022. https://arxiv.org/abs/2212.03533v2

7. Li Z. et al. Towards general text embeddings with multi-stage contrastive learning // arXiv preprint, 2023. https://arxiv.org/abs/2308.03281v1

8. McInnes L., Healy J., Melville J. UMAP: Uniform manifold approximation and projection for dimension reduction // arXiv preprint, 2018. https://arxiv.org/abs/1802.03426v3

9. McInnes L., Healy J., Astels S. hdbscan: Hierarchical density based clustering // The Journal of Open Source Software. 2017. Vol. 2. No 11. Art. no 205.

10. Rousseeuw P. J. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis // Journal of Computational and Applied Mathematics. 1987. Vol. 20. P. 53–65.

11. Caliski T., J Harabasz J. A dendrite method for cluster analysis // Communications in Statistics. 1974. Vol. 3. Iss. 1. P. 1–27.

12. Davies D. L., Bouldin D. W. A cluster separation measure // IEEE Transactions on Pattern Analysis and Machine Intelligence. 1979. Vol. 1. P. 224–227.

13. Chang J. et al. Reading tea leaves: how humans interpret topic models // NIPS'09: Proceedings of the 23rd International Conference on Neural Information Processing Systems. 2009. P. 288–296.

14. Bouma G. Normalized (pointwise) mutual information in collocation extraction // Proceedings of GSCL 30. 2009. P. 31–40.

UDC 004.8

Liu J., Fan J., Davydenko A. A.

Swin-RepGFPN synergized YOLO: A dangerous driving behavior detection in near real-time

1. Abstract. An improved near real-time detection YOLOv11-based model is introduced, addressing high leakage rates and localization inaccuracies in detecting dangerous driving behaviors. By integrating a multi-source dataset of 21,908 images with labels for mobile phone use, drinking, eating, and smoking, we enhance data diversity using clipping, grayscale adjustment, and blurring. The model architecture is refined by incorporating Swin Transformer to improve occlusion detection and lighting adaptation, alongside RepGFPN for optimizing multi-scale feature fusion and small target accuracy via dynamic channel allocation. Experiments show the improved Yolo11-Swin+RepGFPN model achieves an mAP@0.5 of 88.9% (3.3% improvement over baseline) and an mAP@0.5:0.95 of 50.4%, maintaining a real-time speed of 105.5 FPS. Ablation tests reveal that Swin Transformer reduces hand-motion misdetections, while RepGFPN increases recall for small targets like cigarette butts by 6.2%. This solution offers high precision and low latency for in-vehicle safety systems, balancing lightweight design with detection accuracy.

2. Introduction. The advent of advanced driver-assistance systems (ADAS) has brought significant improvements to road safety by enhancing the detection and prevention of dangerous behaviors while driving. However, existing real-time detection models often grapple with high leakage rates and insufficient localization accuracy, particularly in complex driving scenarios where occlusions, variable lighting conditions, and diverse actions pose substantial challenges.

Liu Jie – graduate student, Saint Petersburg State University; e-mail: st122357@student.spbu.ru, phone: +7(981)855-33-39

Fan Jiajie – graduate student, Saint Petersburg State University; e-mail: st122445@student.spbu.ru, phone: +7(963)318-66-59

Davydenko Aleksandr Aleksandrovich – senior lecturer, Saint Petersburg State University; e-mail: a.davydenko@spbu.ru, phone: +7(911)210-57-92

The current body of literature on driver behavior detection primarily focuses on leveraging deep learning architectures [1] like YOLO (You Only Look Once) to achieve fast and accurate object detection. Despite their widespread use, these models often struggle with detecting small or partially obscured objects in dynamic environments. A significant gap remains in enhancing the robustness of such systems against occlusions and lighting variations without compromising real-time performance.

The significance of this study lies in its potential contribution to developing high-precision, low-latency solutions for in-vehicle safety assistance systems. By addressing key challenges associated with current models, this research aims to balance lightweight architecture with enhanced precision, offering valuable insights into optimizing real-time behavior monitoring systems.

3. Dataset. In this study, we integrated multi-source public driving behaviour datasets, cleaned and merged two heterogeneous datasets, and used the Labeling annotation tool to define the four categories of dangerous driving behaviours, namely 'using mobile phone, drinking, eating and smoking', and stored the annotation results in YOLO standard normalized coordinates (`class_id`, `x_center`, `y_centre`, `width`, `height`) to store the annotation results. The generalisation ability is improved by enhancement strategies such as random clipping, horizontal flipping, brightness adjustment, grey scale conversion, Gaussian blurring, pretzel noise, etc., and the results are finally divided into training set (15,335), validation set (4,382), and test set (2,191) according to $7 : 2 : 1$, which provides a highly diversified and accurate database for model training, and supports the demand for behavioural detection in complex driving scenarios. The dataset could be obtained here: https://app.roboflow.com/ljwork.

4. Model architecture. Architecturally, the model introduces two pivotal improvements. Firstly, it replaces the traditional backbone network with a Swin Transformer to leverage hierarchical feature pyramids and moving window self-attention mechanisms, thus improving occlusion handling and adaptation to complex lighting conditions. Secondly, the RepGFPN module is incorporated to optimize multi-scale feature fusion and enhance small target detection accuracy through dynamic channel allocation. The model file is available at https://github.com/LJ343/SR-YOLO

4.1. YOLOv11 benchmark model. YOLOv11 is a single-stage target detection framework proposed by the Ultralytics team in 2024 [2], designed to balance accuracy, speed, and multi-task support. The baseline model architecture in this work comprises the CSPDarknet-C3k2 [3] Backbone, PANet Neck and Lightweight Decoupled Detection Head, enabling multi-task capabilities such as target detection. With an input resolution of 640×640, the model undergoes 4-stage downsampling (step 2 convolution) to generate $P3$ (80×80), $P4$ (40×40), and $P5$ (20×20) feature maps for detecting small, medium, and large targets, respectively. The key processes include: the Backbone extracting features through 4-stage downsampling with C3k2 modules; the Neck aggregating $P3$–$P5$ features using the PANet [4] bidirectional feature pyramid, which fuses information via upsampling and cross-layer connectivity; and the Head independently predicting classification scores, bounding box coordinates, and confidence using decoupled heads. However, the model has two limitations: first, the PANet's feature fusion capability is insufficient for severely occluded targets like a steering wheel occluding a mobile phone, and second, the cross-stage connections in the C3k2 module introduce approximately 12% redundant FLOPs in shallow features.

4.2. Swin Transformer backbone improvement. To improve the model's adaptability to complex driving scenarios (such as occlusion and low light), this work replaces the original CSPDarknet backbone network of YOLOv11 with Swin Transformer [5], and optimizes feature extraction through the hierarchical window attention mechanism and position perception ability. The 640×640 input image is divided into non-overlapping 4×4 blocks by Patch Partition, and multi-scale feature maps are generated through four-stage downsampling. In Stage 1, the output resolution is 80×80 with 96 channels; in Stage 2, it is 40×40 with 192 channels; in Stage 3, it is 20×20 with 384 channels; in Stage 4, it is 10×10 with 768 channels. Moreover, in each stage, resolution is halved and the number of channels is doubled through Patch Merging (splicing adjacent 2×2 blocks and linear projection). The Shifted Window-MSA is adopted, where the feature map is divided into non-overlapping 7×7 windows ($M = 7$) to calculate self-attention and reduce complexity. Adjacent Transformer blocks alternately use regular and shifted windows (shift amount $\lfloor M/2 \rfloor$), and cross-window information interaction is achieved through the formula

$$\mathrm{Shift}(x, y) = \big(x + \lfloor M/2 \rfloor \bmod M,\ \ y + \lfloor M/2 \rfloor \bmod M\big).$$

Cyclic Shift and mask matrix are used to handle non-integer windows after shifting. The Relative Position Bias is also introduced. A learnable relative position encoding matrix B is introduced for each attention head to encode the query-key coordinate difference $(\Delta x, \Delta y)$, with the formula

$$\text{Attention}\,(Q,K,V) = \text{SoftMax}\left(\frac{QK^T}{\sqrt{d}} + B\right)V.$$

Each window shares the B matrix, and only $O\big((2M-1)^2\big)$ parameters are added, avoiding the computational redundancy of global position encoding.

4.3. RepGFPN neck optimisation. To address the feature fusion redundancy in YOLOv11's neck network, this work implements lightweight improvements by referencing DAMO-YOLO's RepGFPN [6] structure: streamlining the Queen-Fusion path by removing 33% redundant upsampling and concatenation operations, retaining only the critical top-down upsampling connections $P5 \rightarrow P4 \rightarrow P3$ while eliminating intermediate repeated concatenation, and adopting direct bottom-up downsampling via stride 2 convolutions to reduce feature dilution.

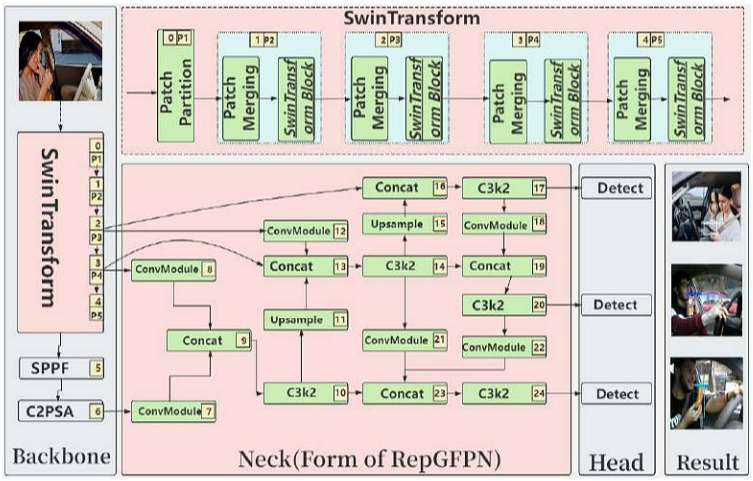


Figure 1. Yolo11-Swin+RepGFPN network architecture diagram

Furthermore, a dynamic channel allocation strategy is applied: the lower-layer $P3$ is set to 256 channels, the middle-layer $P4$ to 512, and

the upper-layer $P5$ to 1024 (reducing 18% FLOPs compared to uniform allocation). Lightweight $C3k2$ modules are enforced after cross-layer concatenation to replace the original ELAN modules, enhancing feature fusion efficiency.

As shown in Fig. 1, the Yolo11-Swin+RepGEPN architecture integrates the Swin Transformer backbone, RepGFPN neck, and lightweight detection head to construct an improved YOLOv11 model with collaborative optimization of hierarchical global attention and a lightweight feature pyramid. This architecture captures long-range spatial dependencies via Swin Transformer, optimizes local detail fusion with RepGFPN to form a global-local feature collaboration mechanism, and dynamically adjusts channel counts and computational intensity based on feature layer semantic depth for adaptive resource allocation. Ultimately, it enables efficient multi-scale ($P3$–$P5$) detection of dangerous driving behaviors at 640×640 input, balancing detection accuracy and computational efficiency in complex scenarios.

5. Experimental evaluation and analysis of results.

5.1. Comparative analysis of key indicators. Table 1 shows the combined performance of the four models in terms of accuracy, efficiency and computational complexity. The specific analyses are as follows:

Table 1. Key metrics

Model	mAP @0.5	mAP @0.5-0.95	GFLOPs	params	Precision	Recall	F1	FPS
Yolo11 (baseline)	85.6%	47.5%	6.6	2.62M	93.1%	95%	0.83	164.9
Yolo11 Swin	84.3%	45.9%	6.1	2.51M	89.5%	93%	0.82	66.3
Yolo11 RepGFPN	88.3%	48.9%	6.8	2.90M	93.4%	94%	0.86	146.6
Yolo11 Swin RepGEPN	88.9%	50.4%	6.3	2.74M	84.8%	95%	0.87	105.5

The improved model Yolo11-Swin+RepGEPN leverages the synergy of Swin Transformer's global attention and RepGFPN's multi-scale fusion, increasing mAP@0.5-0.95 from 47.5% to 50.4% (+2.9%), with Recall reaching 95% to significantly reduce missed detection. Through the lightweight design of the $C3k2$ module, GFLOPs are reduced from 6.6 to 6.3 (−4.5%) while maintaining 105.5 FPS real-time performance,

effectively balancing computational efficiency. Compared with Yolo11-RepGFPN, its parameters are optimized from 2.90M to 2.74M (−5.5%), avoiding model bloat.

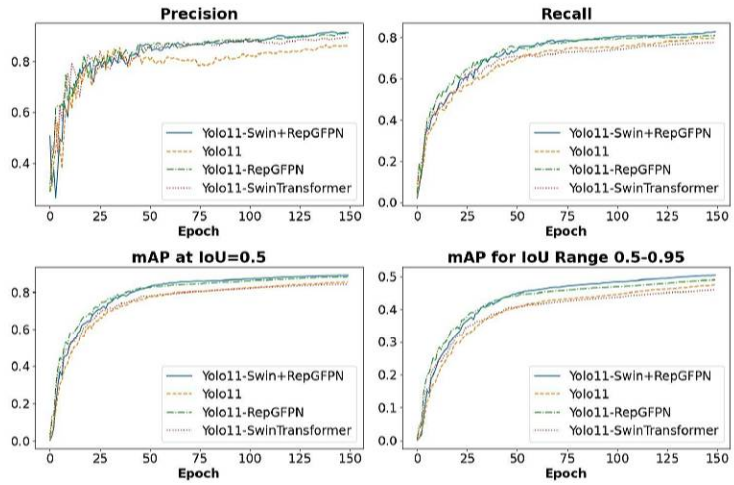


Figure 2. Comparison of the four models on Precision, Recall, and mAP for different IoU standards respectively Trend line graphs

As shown in Fig. 2, in the early stage of training, the Precision of each model fluctuates greatly. In the later stage, Yolo11-Swin+RepGEPN and Yolo11-RepGFPN are more stable with lower false detection rates. In terms of Recall, the two models have stronger detection capabilities in the later stage. In terms of indicators, the mAP@0.5 of Yolo11-Swin+RepGEPN is approximately 0.85, leading in the loose positioning scenario. Under the strict IoU, the two models have obvious advantages, with the mAP approaching 0.5 and more stable positioning. In conclusion, Yolo11-Swin+RepGEPN has better performance.

5.2. Comparative analysis of heat maps. Fig. 3 compares the heatmaps of Yolo11-Swin+RepGEPN with other versions, demonstrating stronger object perception capabilities: Yolo11-Swin+RepGEPN exhibits clearer object contours and more precise foreground activation. One can see the model doesn't detect non-significant artifacts (like door parts, glass, etc.) and tend to focus on subjects only.

This is attributed to the synergy of Swin Transformer capturing holistic hand postures (coherent heat regions) and RepGFPN enhancing edges and details.



Figure 3. Heat maps of Yolo11 (baseline), Yolo11-Swin, Yolo11-RepGFPN vs. Yolo11-Swin+RepGFPN.

6. Conclusion. This study proposes an improved YOLOv11 model that integrates Swin Transformer and re-parameterized feature pyramid. Through the collaborative optimization of global attention mechanism and dynamic multi-scale feature fusion, the accuracy and robustness of dangerous driving behavior detection are significantly enhanced. Experiments show that the improved model exhibits more stable convergence characteristics and generalization ability in complex scenarios (such as occlusion, weak illumination), and its recall rate and comprehensive detection performance are significantly superior to those of traditional single-stage detectors. Although there is a slight decrease in inference speed, the model maintains the efficiency balance required for on-board real-time monitoring through the calculation path optimization and channel dynamic allocation strategies. Future research will focus on

lightweight module design and multimodal data fusion to further improve the practicality and generalization performance of the model in extreme scenarios, providing more reliable technical support for intelligent driving safety systems [7].

References

1. Wang C., Lin M., Shao L., Xiang J. MF-YOLO: A Lightweight Method for Real-Time Dangerous Driving Behavior Detection // IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2024.

2. Khanam R., Hussain M. YOLOv11: An Overview of the Key Architectural Enhancements // `https://arxiv.org/abs/2410.17725`. 2024.

3. Wang C. Y., Bochkovskiy A., Liao H. Y. M. YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors // IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2023. P. 7469–7478.

4. Wang X., Girshick R., Gupta A. et al. Non-local Neural Networks // IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2018. P. 7794–7803.

5. Liu Z., Lin Y., Cao Y. et al. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows // IEEE/CVF International Conference on Computer Vision (ICCV). 2021. P. 10012–10022.

6. Xu X., Jiang Y., Chen W. et al. DAMO-YOLO: A Report on Real-Time Object Detection Design // European Conference on Computer Vision (ECCV). 2022. P. 112–128.

7. Tian X., Wang Q., Yang J. Z. et al. Optimization design of MnO2-based fiber supercapacitors with diverse crystal structures // Journal of Power Sources. 2025. No 632. P. 236377.

UDC 004.02

Wang Q., Gao Z., Zhang L.

Research on tomato maturity detection model based on YOLOV11 in complex lighting scenarios

Recommended for publication by associate professor Blekanov I. S.

1. Introduction. Tomato is the most widely grown fruit and vegetable crop in the world. According to FAO data for 2023, it produces 182 million tonnes annually. The fruit is rich in key nutrients such as lycopene, vitamin C and potassium [1]. Global consumption of tomato products is growing at a rate of 4.7 percent per year as healthy diets become more popular, fuelling innovations in tomato cultivation technology. The development of photo-selective agro-materials technology has reshaped the cultivation of protected tomatoes by enabling the light quality regulation of shade nets. It was found that the use of red shade nets resulted in a 21 percent [2] increase in total tomato yield, a reduction in fruit cracking from 17.3 percent to 4.1 percent under full-light conditions, and a significant increase in lycopene content of 34.9 percent compared to open field cultivation. However, tomato colour varies at different growth stages due to differences in light quality and composition. Existing ripening detection models have drastically decreased in accuracy, cannot effectively monitor the semi-ripe state, and are prone to tomato rot losses. To address this problem, we develop a YOLO-based high-precision tomato ripeness detection model for complex lighting scenarios, which improves the detection of the three ripening states of tomatoes without loss of efficiency.

2. Dataset.

2.1. Dataset overview. The "Tomato Plant Factory Dataset" [3] was collected in the complex lighting environment of an artificial light plant factory, including complex situations such as perspective, lighting

Wang Qi – graduate student, Saint Petersburg State University; e-mail: st122208@student.spbu.ru, phone: +7(981)803-52-22

Gao Zihan – graduate student, Saint Petersburg State University; e-mail: st122237@student.spbu.ru, phone: +7(981)791-11-94

Zhang Li – graduate student, Saint Petersburg State University; e-mail: st122491@student.spbu.ru, phone: +7(981)789-31-11

quality, distance changes, and fruit occlusion and blurring. In this study, the dataset was optimized for the second time. Incorrect categories were corrected, and a "semi-mature" annotation category was added. Tomatoes are divided into three states: mature, semi-mature, and immature. Immature fruits begin to expand, and their epidermis is light green more than three quarters of the surface of semi-mature fruits turns red or yellow; mature fruits have completely red and shiny skins. The dataset contains a total of 9,112 instances, including 5,996 green fruits and 3,116 red fruits. It is divided into a training set, a validation set, and a test set in a ratio of 8:1:1. The training set has 416 images, and the validation set and the test set each have 51 images. Sample data of cherry tomatoes under different lighting conditions, shooting angles, and maturities are shown in Figure 1.



Figure 1. Examples from the dataset

2.2. Data augmentation. To improve the network training effect and the model's generalization ability, data augmentation was performed on the 416 images in the training set. This included horizontal and vertical flips, random adjustments of brightness and contrast, and simulations of image blurring effects caused by fog, rain, and camera defocus. Extending the training set to 4979 images. Some of the enhanced images are shown in Figure 2.



Figure 2. Examples of data augmentation

3. Model and evaluation methods.

3.1. YOLOv11 model overview. YOLOv11 [4] is a popular detection model currently. It adopts an improved convolutional neural network architecture and combines multiscale feature extraction technology, enabling it to detect objects of different sizes and shapes. In addition, due to the adoption of more advanced loss functions and optimization algorithms, YOLOv11 has significantly improved performance in object localization and classification. The model divides the input image into fixed-size grids and simultaneously predicts the bounding boxes and the probabilities of the corresponding categories in each grid, achieving end to end training and inference, thus reducing the complex processes of region extraction and classification in traditional detection methods. It is applicable to various scenarios such as autonomous driving, video surveillance, and industrial inspection.

3.2. Model evaluation methods. In this study, we use the most widely applied assessment method: mAP assessed the model's target identification and localization across different recall rates, reflecting complex scene performance. Precision (P) showed the proportion of true positives among predicted targets, recall (R) the proportion of actual targets detected, and the F1 score balanced P and R for comprehensive detection precision and coverage, especially with imbalanced samples. Higher metric values indicated better performing models.

4. Experimental methods and results.

4.1. Methods of experimental. This study made two key structural modifications to the YOLOv11 model: First, the standard convolution in the backbone network was replaced with PConv [5] (windmill shaped convolution). The original 33 convolution was decomposed into a local feature extraction kernel and a global feature fusion kernel through the windmill shaped weights, reducing the computational complexity and the number of parameters while maintaining the model accuracy, and using a dynamic grouping strategy to balance efficiency and performance. Second, the ASFFHead [6] detection head was used to replace the original detection module. The feature pyramid was reconstructed through three level adaptive weight fusion (ASFF 3). After bilinear interpolation alignment of the feature maps, a lightweight weight network was used to generate a channel attention map, and pixel level feature calibration was achieved through Softmax normalization, effectively solving the cross scale feature conflict problem and improving the detection

performance. The improved network structure is shown in Figure 3.

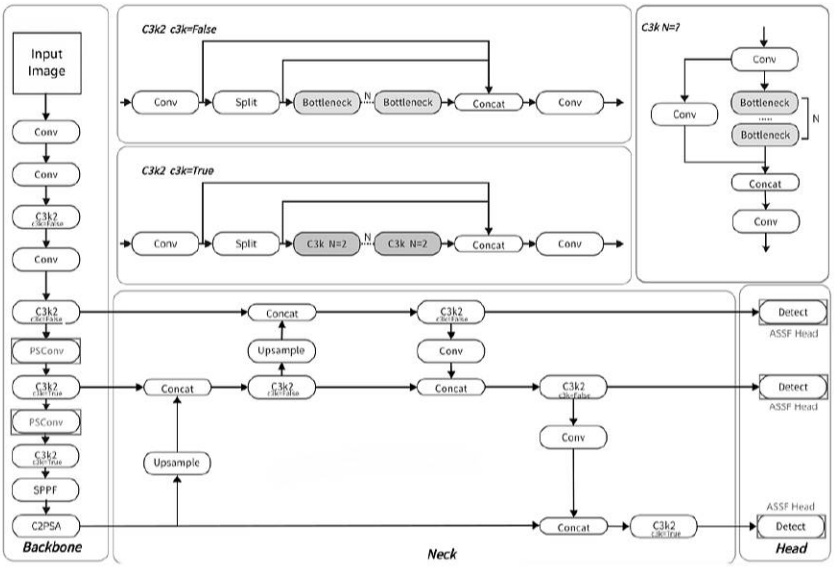


Figure 3. model network structure

4.2. Results of ablation experiments. An ablation experiment was carried out to compare the performance of different models. The results are shown in table.

Models	Box(P)	R	mAP50	mAP50-95	F1
PConv	0.829	0.826	0.852	0.667	0.81
ASSF	0.835	0.803	0.837	0.667	0.81
Yolov11n	0.846	0.773	0.84	0.646	0.81
PCAF	0.833	0.803	**0.859**	**0.668**	**0.82**

Table. Results of ablation experiments

The experimental results show that compared with the original YOLOv11 model, the improved model has 1.9% increase in mAP50, 2.2% increase in mAP50-95, and 1% increase in F1.

5. Conclusion. This study developed a detection model based on YOLO for the problem of tomato maturity detection in complex lighting scenarios. Through dataset optimization, data augmentation methods, and model structure improvement, the detection accuracy and efficiency of the model for tomato maturity in complex lighting environments were effectively improved, and the semi-mature stage could be accurately identified. This model provides reliable technical support for intelligent tomato cultivation and is expected to play an important role in actual production. Future research can further explore more optimized model structures and algorithms to adapt to more complex planting environments and detection requirements.

References

1. Perveen R., Suleria H. A. R., Anjum F. M., et al. Tomato (solanum lycopersicum) carotenoids and lycopenes chemistry; metabolism, absorption, nutrition, and allied health claims – A Comprehensive Review // Critical Reviews in Food Science and Nutrition. 2015. Vol. 55. No 7. P. 919–929.

2. Ilić Z. S., Milenković L., Stanojević L., et al. Effects of the modification of light intensity by color shade nets on yield and quality of tomato fruits // Scientia Horticulturae. 2012. Vol. 139. P. 90–95.

3. Wu Z., Liu M., Sun C., et al. A dataset of tomato fruits images for object detection in the complex lighting environment of plant factories // Data in Brief. 2023. Vol. 48. Art. no 109291.

4. Khanam R., Hussain M. YOLOv11: an overview of the key architectural enhancements // ArXiv preprint, 2024. https://arxiv.org/abs/2410.17725v1

5. Yang J., Liu S., Wu J., et al. Pinwheel-shaped convolution and scale-based dynamic loss for infrared small target detection // ArXiv preprint, 2024. https://arxiv.org/abs/2412.16986v1

6. Liu S., Huang D., Wang Y. Learning spatial fusion for single-shot object detection // ArXiv preprint, 2019. https://arxiv.org/abs/1911.09516v2

UDC 004

Xu H., Petrosian O. L.

Printed circuit board defect detection based on improved RT-DETR

1. Introduction. The main defect detection methods used today include traditional methods and deep learning.

Traditional methods mainly include manual identification methods, feature extraction techniques, and machine learning algorithms. The manual identification method is inefficient and labor intensive, with low precision and poor real-time performance [1]; Machine learning algorithms based on feature extraction [2] are one of the most commonly used ways to detect surface defects. Ahmad et al. [3] combined feature extraction methods with classifiers such as Support Vector Machines to achieve defect identification.

Popular deep learning models include Fast RCNN [4], Faster RCNN [5], SSD [6], YOLO [7], etc. In recent years, the rise of Transformers has had a profound impact on the field of deep learning computer vision. Carion et al. [8] introduced DETR (Detection Transformer), an end-to-end object detection network based on Transformers.

To achieve real-time detection, Zhao et al. [9] proposed RT-DETR. RT-DETR eliminates the effect of non-maximum suppression and outperforms the SOTA detector YOLO in terms of speed and accuracy. It solves the problems of high computational cost and difficult-to-optimise object queries that exist in existing DETR models. RT-DETR consists of three parts: a backbone, an Efficient Hybrid Encoder (EHE) and a Transformer decoder with auxiliary prediction heads.

2. GSSF-RT-DETR-WPIoU.

2.1. Efficient Hybrid Encoder with GSConv. EHE consists of two modules, namely the Attention-based Intra-scale Feature Interaction (AIFI) and the CNN-based Cross-scale Feature Fusion (CCFF).Compared with directly splicing multi-scale features into Transformer encoder, CCFF reduces computational redundancy through con-

Xu Hongfan – graduate student, Saint Petersburg State University; e-mail: st115838@student.spbu.ru, phone: +7(906)226-89-75

Petrosian Ovanes Leonovich – professor, Saint Petersburg State University; e-mail: o.petrosyan@spbu.ru, phone: +7(911)740-80-19

volution operation. The core component of CCFF is the Fusion module, which is used to fuse multi-scale features. The number of parameters and calculation amount of CCFF are large. In this study, a lightweight and efficient hybrid encoder is proposed. Its structure is shown in Figure 1.

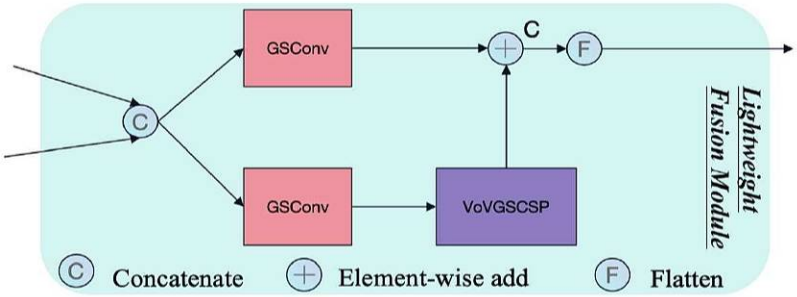


Figure 1. Lightweight fusion module

GSConv and VoVGSCSP are introduced to replace the original convolution and RepConv in Fusion respectively to build a lightweight CCFF. The computational cost of GSConv is 50% of that of traditional convolution.

2.2. Scale Sequence Feature Fusion (SSFF). The SSFF module solves the limitations of traditional feature pyramid networks by effectively fusing multi-scale features to deal with objects with different sizes, orientations and aspect ratios. The structure of the SSFF is shown in Figure 2.

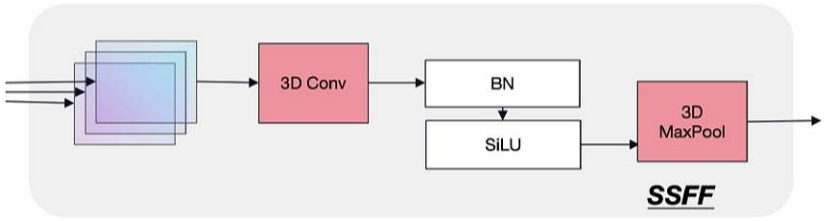


Figure 2. Scale Sequence Feature Fusion module

In the improved network, the SSFF receives P3, P4, and P5 feature maps processed by GSConv, and adjusts the number of channels

to align the feature maps at different scales; The 3D convolution layer is used to further process the adjusted feature map. Normalization and nonlinear transformation by BatchNorm(BN) and activation function. Finally, through the 3D pooling operation, the feature graph is reduced to retain the most important feature information.

2.3. WPIoU. Intersection over Union (IoU) is a commonly used evaluation metric in object detection tasks. It measures how much the predicted bounding box overlaps with the ground truth bounding box. This study improves the IoU of RT-DETR, combines Wise-IoU and Powerful IoU, and constructs WPIoU loss as the loss function of the improved algorithm.

PIoU enhances the sensitivity of the corner alignment between the prediction box and the ground truth by introducing the position deviation P. The formula of PIoU Loss is as follows:

$$\text{PIoU Loss} = \text{IoU Loss} - e^{-P^2} + 1,$$

The positional deviation P is used to measure the difference between the predicted box and the ground truth in terms of the position of the corner points, with the formula

$$P = \frac{1}{4}\left(\frac{dw_1 + dw_2}{|w_2|} + \frac{dh_1 + dh_2}{|h_2|}\right),$$

d_{w1} and d_{w2} represent the angular deviation of the prediction box and the ground truth in the x direction, respectively; d_{h1} and d_{h2} represent the angular deviation of the prediction box and the ground truth in the y direction, respectively; w_2 and h_2 are the width and height of the ground truth.

WIoU is a bounding box regression loss function based on Dynamic Non-Monotonic Focusing Mechanism. WIoU further improves the loss function of the original model based on PIoU with the following formula:

$$\text{WPIoU Loss} = \frac{\beta}{\delta \cdot \alpha^{\beta-\delta}} \cdot \text{PIoU Loss},$$

where,

$$\beta = \frac{\text{IoU Loss}}{\text{Iou Mean}} \in [0, +\infty],$$

β is the outlier degree of the anchor frame. IoU Mean is the dynamically updated IoU loss mean. α and δ are hyper parameters. The dynamic

loss scaling mechanism can adaptively adjust the loss weights during the training process, avoiding the gradient disappearing or exploding at the late stage of training, and improving the convergence speed and stability of the model.

3. Experiment. The PCB dataset is published by Peking University. PCB defects are shown in Figure 3: short circuit, spur, rat bite, open circuit, leakage hole, false copper.

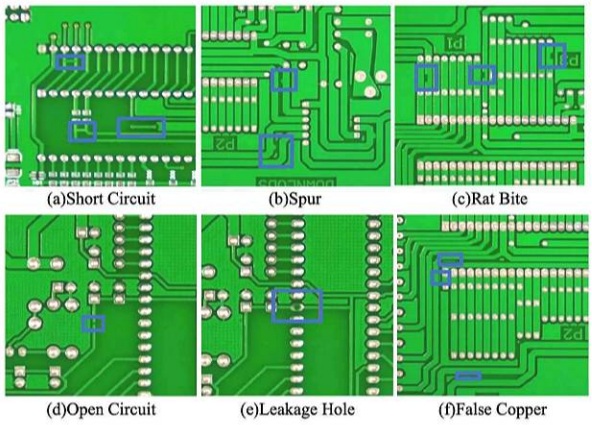


Figure 3. Types of defects

The epoch of all models is set to 150 in training. Divide the dataset into training set, validation set, test set in the ratio of 7:2:1. The backbone of RT-DETR is ResNet18, and the improved model is GSSF-RT-DETR-WPIOU (hereinafter referred to as modelG). The quality metrics changes in the training are shown in Figure 4. The experimental results show that RT-DETR model is superior to YOLO model in overall performance. modelG's performance is significantly better than other models, which reflects the effectiveness of the improvement strategy.

The experimental results are shown in Table. The precision(P), recall(R) and mAP50 of ModelG are the highest. Compared with RT-DETR, the precision of ModelG is improved by 5.2%, the recall by 0.7%, and the mAP50 by 0.8%.

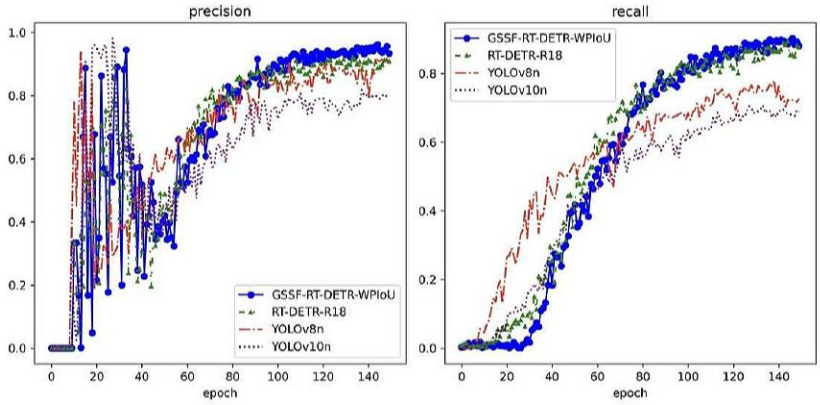


Figure 4. Experimental results on the validation set

Table. Experimental results of different models on PCB dataset

	P (%)	**R (%)**	**mAP50 (%)**
YOLOv8n	89.7	76.9	84.1
YOLOv10n	79.8	68.0	76.5
RT-DETR	90.4	88.6	92.1
ModelG	95.6	89.3	92.9

ModelG has the best precision on all indexes and can maintain high recall and mAP50 indicating that its model generalization and detection ability are strong. The RT-DETR is more suitable for PCB dataset than the YOLO series, and its higher mAP50 indicate that it is more stable for object detection tasks.

4. Summary and outlook. This study presents a lightweight improvement to the RT-DETR, optimizing both accuracy and efficiency for defect detection in industrial applications. Looking ahead, future research can focus on further optimizing computational efficiency to enable real-time deployment in industrial environments. Applying the improved RT-DETR framework to broader industrial scenarios beyond PCB inspection may further validate its versatility and effectiveness.

References

1. Jha S. B., Babiceanu R. F. Deep CNN-based visual defect detection: Survey of current literature // Computers in Industry. 2023. Vol. 148. P. 103911.

2. Zheng X. et al. Recent advances in surface defect inspection of industrial products using deep learning techniques // The International Journal of Advanced Manufacturing Technology. 2021. Vol. 113. P. 35–58.

3. Ahmad A. et al. Photovoltaic cell defect classification using convolutional neural network and support vector machine // IET Renewable Power Generation. 2020. Vol. 14. No 14. P. 2693–2702.

4. Girshick R. Fast R-CNN // Proceedings of the IEEE International Conference on Computer Vision. 2015. P. 1440–1448.

5. Ren S. et al. Faster R-CNN: Towards real-time object detection with region proposal networks // IEEE Transactions on Pattern Analysis and Machine Intelligence. 2016. Vol. 39. No 6. P. 1137–1149.

6. Liu W. et al. SSD: Single shot multibox detector // Computer VisionECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I. Springer International Publishing. 2016. P. 21–37.

7. Redmon J. et al. You only look once: Unified, real-time object detection // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016. P. 779–788.

8. Carion N. et al. End-to-end object detection with transformers // European Conference on Computer Vision. Cham: Springer International Publishing. 2020. P. 213–229.

9. Zhao Y. et al. DETRs beat YOLOs on real-time object detection // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2024. P. 16965–16974.

UDC 004

Zhang L., Wu H., Petrosyan O. L.

Online adaptive forecasting for web traffic with sudden drift

1. Introduction. With the development of science and technology and the increase in the scale of streaming data, how to effectively process and analyze these massive and changing data has become an important topic in scientific research and practical applications. Traditional modeling and prediction methods often face many challenges.

On the one hand, traditional models based on batch learning (such as ARIMA [1] and linear regression [2]) rely on training with the full amount of historical data and cannot adapt to the continuous arrival characteristics of data streams. Frequent retraining leads to high computing costs and decision delays.

On the other hand, concept drift is prevalent in streaming data [3], which causes data distribution to change over time. Conventional models cannot adapt to data changes in real time, thereby reducing the prediction performance of static models. Considering the high-dimensional and massive characteristics of streaming data, in [4] the authors proposed an optimal sampling strategy based on clustering patterns, which extracts representative samples from the data and significantly improves the efficiency of concept drift detection.

In streaming data modeling, multi-step prediction [5] can effectively improve the ability and robustness of the model and has become an indispensable technology. Multi-step prediction can capture the long-term dependency of the time series by generating prediction results for multiple future time points at the same time, smooth the impact of short-term fluctuations on the model, and thus enhance robustness.

Online learning methods provide an effective solution for streaming data modeling. Its core value lies in breaking through the spatiotemporal

Zhang Lili – graduate student, Saint Petersburg State University; e-mail: st115843@student.spbu.ru, phone: +7(906)226-91-07

Wu Haitao – PhD student, Saint Petersburg State University; e-mail: wuhaitao2023@gmail.com, phone: +7(996)786-53-78

Petrosian Ovanes Leonovich – professor, Saint Petersburg State University; e-mail: o.petrosyan@spbu.ru, phone: +7(911)740-80-19

constraints of traditional batch learning and building a closed-loop response system from data flow to model update. In [6] an online adaptive learning framework is proposed that combines clustering for resampling and retrains the model in time after detecting concept drift. This effectively solves the concept drift problem of streaming data, but the accuracy of the model will still decrease when the algorithm faces sudden drift. Based on Haitao's adaptive framework, this paper introduces a data annotation mechanism, which greatly reduces the impact of sudden drift and improves the performance of the algorithm.

2. Data acquisition and processing. The data can be obtained from the "Web Traffic Time Series Forecasting" competition on the kaggle website. Each time series represents the daily page views of different Wikipedia articles from July 1, 2015 to December 31, 2016. The data file contains variables such as date, website id, way to enter the website, website category, and website page views. This paper first deletes missing values, and then screens the daily page views of the five most popular web pages from eight countries (such as France, the United Kingdom, etc.) as instance data for this study. This dataset reflects the network access patterns of users in different countries and can provide us with deep insights into website traffic and user behavior. Through these real data, we can effectively evaluate the proposed online learning algorithm and analyze its performance and adaptability in practical applications.

3. Online adaptive algorithm. In this section we describe the proposed algorithm that uses the resampling and the adaptive framework.

3.1. Resampling. A clustering-based resampling method is introduced in detail. Historical data similar to recent data is obtained and combined with recent data to provide a high-quality data foundation for online adaptation. The specific process is shown in the figure below.

By using a sliding window approach, the data is simulated as streaming data, and an online algorithm is introduced. Based on the correlation coefficients between variables, the feature variables that are important for the clustering algorithm are manually selected. A dimensionality reduction model is trained using historical data. MiniBatchKMeans is applied to cluster the historical data, returning cluster labels N. The trained dimensionality reduction model is used to reduce the dimensionality of recent data, and the clustering model is applied to the recent data to partition it into c clusters. The corre-

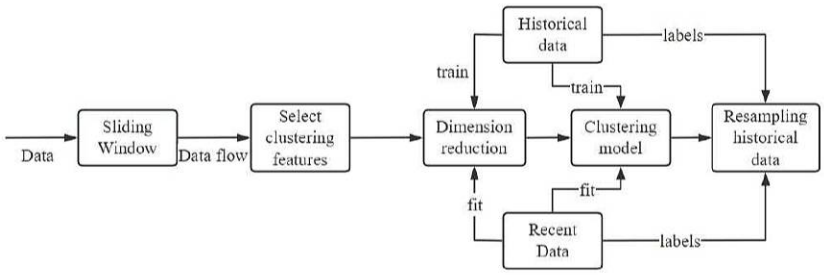


Figure 1. Resampling flow chart

sponding cluster labels M and sample counts are obtained, denoted as $\{M_1 : m_1, M_2 : m_2, \ldots, M_c : m_c\}$, where M_i represents the cluster label in recent data and m_i is the sample count of the corresponding cluster, with $m_1 > m_2 > \ldots > m_c$. From the historical data cluster labels N, the recent data cluster labels M and their corresponding historical sample counts are extracted, denoted as $\{M_1 : n_1, M_2 : n_2, \ldots, M_c : n_c\}$, where n_i represents the sample count of cluster M_i in historical data.

The cluster label M_1 with the largest sample count in recent data is selected, and all samples n_1 from the corresponding cluster in historical data are directly included in the sampling. For the remaining cluster labels $(M_2, \ldots, M_c)$, samples are proportionally selected. The minimum sample count $a = \min(n_2, \ldots, n_c)$ from historical data is used as the maximum sampling size for other clusters. The sampling base b_i for each remaining cluster is determined based on the proportion of recent data samples m_i to the total recent data samples $\text{len}(y_r)$:

$$b_i = a\frac{m_i}{\text{len}(y_r)}, \quad i = 2, \ldots, c.$$

This cluster-based resampling method ensures that the resampled historical data distribution aligns more closely with the recent data distribution, effectively utilizing drift information in recent data and preventing model overfitting.

3.2. Online adaptive framework. This paper proposes a multi-step forecasting model to provide regression model support for online adaptive algorithms. Assuming the forecasting horizon is H, X_t represents the explanatory variables, y_t is the response variable, l denotes the historical timestamps, ε_h is the residual normally distributed in the fore-

cast, and f_h is the LightGBM model established for h-step forecasting. The model inputs include historical values of X_t and y_t. For the h-th prediction layer regression model ($h \in \{1, \ldots, H\}$), the formulation is:

$$\mathbf{y}_{t+h} = f_h\left(X_t, \ldots, X_{t+l}, y_t, \ldots, y_{t+l}\right) + \varepsilon_h.$$

In order to solve the problem of response delay of traditional adaptive methods in the scenario of sudden drift, this paper proposes an innovative data labeling mechanism. Starting from the data itself, a reasonable range of response variable values is proposed. According to whether the response variable value exceeds the range, a Boolean discriminant factor is introduced to dynamically mark the change of data distribution, thereby improving the detection and adaptive ability of the algorithm to sudden drift. Most existing adaptive algorithms use accuracy to detect concept drift. This paper uses the mean absolute error value as the detection condition for concept drift, making the model more sensitive to concept drift detection and achieving higher accuracy.

The online adaptive algorithm in this paper is an online and offline dual-mode algorithm. The offline multi-step regression model is established using the first 20 percentage of the data, and the sliding window method is used to simulate the data stream. As the stream data is traversed, the predicted value is obtained through the model, and the concept drift detection condition (MAE) is used to determine whether concept drift occurs. If the concept drift condition is not met, it means that the offline model does not need to be changed; if concept drift is detected, the data is labeled and resampled, and the model is updated with the processed data to maintain the accuracy and robustness of the model prediction. After adding the data labeling mechanism, the algorithm speeds up the response to mutation drift. The framework diagram of the online adaptive algorithm in this article is as follows:

4. Results. Applying the online adaptive algorithm with data annotation function proposed in this paper, the values of R2 and MAE are calculated as follows:

Table. Comparison of performance indicators of different models

	R2	**MAE**
Replacement(Sudden)	0.929	0.189
Replacement [6]	0.912	0.191
OfflineLightGBM [5]	0.747	0.192

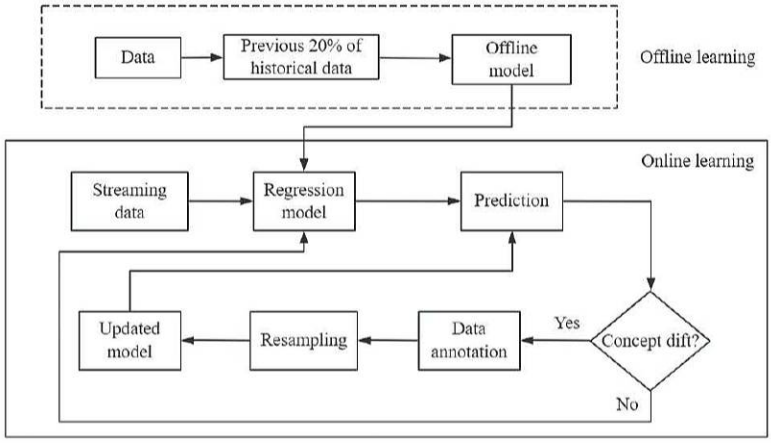


Figure 2. Online Adaptive Framework Diagram

Traversing the streaming data, at each timestamp, the R-squared change curves of offline learning (solid line), online learning algorithm (this method) (dotted line), and online learning algorithm without labeling mechanism (dash-dot line) are shown in the figure 3.

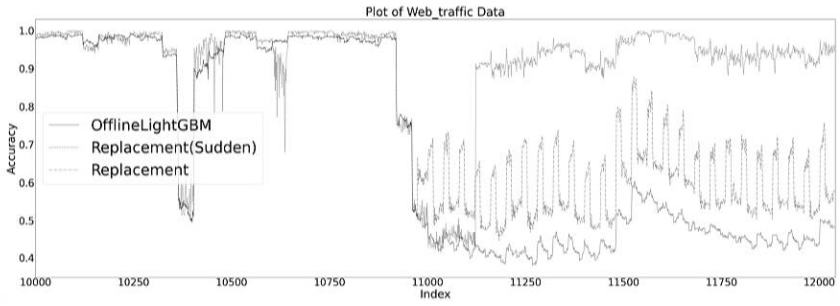


Figure 3. Accuracy plot of different models under sudden drift

Whether through the indicator table or the curve chart, we can see that the improved method in this paper (dotted line) can effectively deal with sudden drift, which not only improves the model accuracy but also reduces the model's reaction time. The algorithm in this paper performs best and is suitable for streaming data with sudden drift.

References

1. Ho S. L., Xie M. The use of ARIMA models for reliability forecasting and analysis // Computers & Industrial Engineering. 1998. Vol. 35. Iss. 1–2. P. 213–216.

2. Nathans L. L., Oswald F. L., Nimon K. Interpreting multiple linear regression: a guidebook of variable importance // Practical Assessment, Research, and Evaluation. 2012. Vol. 17. No 9. Art. no 9.

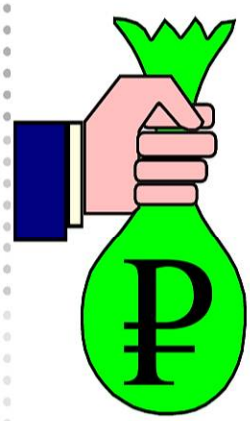
3. Gama J., Žliobaitė I., Bifet A., et al. A survey on concept drift adaptation // ACM Computing Surveys (CSUR). 2014. Vol. 46. No 4. Art. no 44.

4. Ren H., Zou C., Chen N., Li R. Large-scale datastreams surveillance via pattern-oriented-sampling // Journal of the American Statistical Association. 2022. Vol. 117. No 538. P. 794–808.

5. Taieb S. B., Bontempi G., Atiya A. F., Sorjamaa A. A review and comparison of strategies for multi-step ahead time series forecasting based on the NN5 forecasting competition // Expert Systems with Applications. 2012. Vol. 39. No 8. P. 7067–7083.

6. Wu H., Dolgintseva E., Zhadan A., Petrosian O. Forecasting online adaptation methods for energy domain // Engineering Applications of Artificial Intelligence. 2023. Vol. 123. Art. no 106499.

5. Управление социально-экономическими системами



УДК 339.372+517.977.5

Ван Д., Смирнов Н. В., Го Ч.

Модель оптимального управления логистикой торговой фирмы

1. Введение. Настоящее исследование основывается на методах и подходах решения задач оптимального управления, предложенных в [1,2]. Также используется опыт авторов работ [3,4], в которых предложены модель системы производства скоропортящихся продуктов и стратегии оптимального управления с учетом изменения уровней качества продуктов во времени.

В данной статье предложена динамическая модель торгового предприятия, реализующего скоропортящиеся товары. На основе этой модели сформулирована задача оптимального управления оптовыми закупками с целью максимизации прибыли и минимизации издержек за счет балансировки избытка и недостатка товара.

Разработанный алгоритм способен вычислять оптимальную стратегию, постоянно обновляя и корректируя скорость заказов на основе фактического уровня текущих запасов. Тестирование алгоритма на реальных данных китайской компании по продаже бананов показало, что оптимальная стратегия превосходит по прибыли существующие стратегии продавцов. В заключении указаны направления развития модели.

2. Описание модели и синтез оптимального управления. Рассмотрим оптимизацию процесса заказа и продаж скоропортящихся продуктов за $N - j$ дней, предполагая немедленную доставку оптовых заказов. В конце каждого рабочего дня розничный торговец сортирует оставшийся товар и принимает свой очередной заказ. Предположим, что начальный объем товара в день $k \in [j, N]$ равен $x(k-1)$, где j – начальный день. Предполагаемый объем продаж в

Ван Дань – аспирант, Санкт-Петербургский государственный университет; e-mail: st102920@student.spbu.ru, тел.: +7(812)363-60-00 (9553)

Смирнов Николай Васильевич – профессор, Санкт-Петербургский государственный университет; e-mail: n.v.smirnov@spbu.ru, тел.: +7(812)363-60-00 (9553)

Го Чжао – аспирант, Санкт-Петербургский государственный университет; e-mail: st097122@student.spbu.ru, тел.: +7(812)363-60-00 (9553)

день k зависит от доли продаж $s_c(k)$, т. е.

$$D(k) = s_c(k)x[k-1] = \left(\frac{c_o}{p(k)}\right)^{\beta} x(k-1),$$

где $c_o(k)$ – стоимость закупки единицы товара в день k, $p(k)$ – цена продажи в день k, β – коэффициент аппроксимации.

Учитывая, что скорость порчи скоропортящихся продуктов зависит от погоды, сезона и времени, обозначим ее через $a(k)$, соответственно в день k. Тогда в конце каждого рабочего дня остаток запасов составит $x_m(k) = (1-a(k))x(k-1) - (1-a(k))D(k)$. После окончания рабочего дня торговец заказывает новый объем товара $u(k)$. Тогда начальный объем на следующий день $x(k)$ будет

$$x(k) = x_m(k) + u(k) = (1-a(k))(1-s_c(k))x(k-1) + u(k). \qquad (1)$$

На самом деле, под влиянием различных возмущающих факторов, предполагаемый остаток запасов $x_m(k)$ и фактический запас $x_r(k)$ различны. Обозначим $w(k)$ разность/ошибку между ними.

Для уменьшения влияния ошибки на стратегию заказа, когда процесс продаж достигает дня $j = k$, мы заменяем величину $x_m(j)$ на фактически остаток $x_r(j)$. Затем, используя динамическое уравнение (1), вычисляем будущие остатки запасов с дня $k+1$ по день N и на их основе разрабатываем оптимальную стратегию заказов $U = [u(j), u(j+1), \ldots, u(N)]$. Таким образом, динамическое уравнение (1) можно записать так:

$$\begin{cases} x(k) = x_r(k) + B(k)u(k), & k = j, \\ x(k) = A(k)x(k-1) + B(k)u(k), & k > j, \end{cases} \qquad (2)$$

где $A(k) = (1-a(k))(1-s_c(k))$, $B(k) = 1$, $j \in [k, N]$ – количество дней, прошедших с начала продаж, а $k \in [1, N]$ – этап планирования.

Общее решение уравнения (2) представимо в виде

$$x(k) = M_j(k)U_j(k) + \eta_j(k), \quad k \in [j, N], \qquad (3)$$

где

$$M_j(k) = \begin{pmatrix} A(k) \times \ldots \times A(j+1)b(j) & \ldots & A(k)b(k-1) & b(k) \end{pmatrix},$$

$U_j(k) = (u(j), \ldots, u(k))^T$ – вектор управления размерности k, $\eta_j(k) = A(k) \times \ldots \times A(j+1)x_r(j)$ – скаляр.

Далее рассмотрим функцию прибыли коммерческого предприятия. Прибыль состоит из нескольких частей: 1) ежедневный доход от продаж $p(k)D(k)$; 2) доход от продажи испорченных товаров по сниженной цене $p_d a(k)(x(k-1)-D(k))$, где p_d – цена продажи единицы испорченного товара; 3) затраты, включая стоимость хранения $c_s x(k-1)$ и стоимость закупки свежих товаров $c_o u(k)$, где $c_s > 0$ – затраты на единицу хранения. Тогда целевая функция в день k выражается как

$$r(k) = \big(p(k)s_c(k) + p_d a(k)(1 - s_c(k)) - c_s\big)x(k-1) - c_o u(k).$$

Используя формулу общего решения (3), целевую функцию можно выразить так:

$$\begin{cases} r(k) = -c_o u(k), \quad k = j, \\ r(k, U_j(k)) = Q(k)U_j(k) + \widetilde{\eta}_j(k), \quad k > j, \end{cases} \tag{4}$$

где

$$\begin{aligned} Q_j(k) &= \big(\big(p(k)s_c(k) + p_d a(k)\big(1 - s_c(k)\big) - c_s\big)\, M_j(k-1),\ -c_o\big)\,, \\ \widetilde{\eta}_j(k) &= \big(p(k)s_c(k) + p_d a(k)\big(1 - s_c(k)\big) - c_s\big)\, \eta_j(k-1). \end{aligned}$$

Здесь $Q_j(k)$ объединяет все факторы, влияющие на прибыль со дня j по день k, а $\widetilde{\eta}_j(k)$ отражает часть, независящую от управления. Таким образом, цель в течение планового периода со дня j по день N заключается в максимизации кумулятивной прибыли, а целевая функция планового периода выражается на основе (4) следующим образом:

$$\sum_{k=j}^{N} r(k, U_j(k)) = \sum_{k=j}^{N} \Big(Q_j(k)U_j(k)\Big) \to \max, \tag{5}$$

$$\begin{aligned} &Q_j = \big(Q(j), 0, \ldots, 0\big) = \big(-c_o, 0, \ldots, 0\big), \quad k = j, \\ &Q_k = \big(Q(k), 0, \ldots, 0\big) = \big(\gamma(k)M_j(k-1), -c_o, 0, \ldots, 0\big), \quad k > j, \\ &\ldots\ldots \\ &Q_N = Q(N) = \big(\gamma(N)M(N-1), -c_o\big), \\ &Q_{S_j}(N) = \textstyle\sum_{k=j}^{N} Q_k, \quad k \in [j, N], \end{aligned}$$

где скаляр $\gamma(k) = p(k)s_c(k) + p_d a(k)(1 - s_c(k)) - c_s$. В каждой строке Q_k подстрока нулей содержит $N - k$ нулей для $k \in [j, N]$, $j \in [1, N]$.

Таким образом, (5) можно записать в следующем виде:

$$Q_{S_j}(N)U_j(N) \to \max. \tag{6}$$

Далее рассмотрим ограничения на фазовые переменные и естественные экономические ограничения на управление. Во-первых, это ограничение емкости склада. В любой день должно выполняться неравенство:

$$0 \leqslant x(k) \leqslant g, \quad k = j, \ldots, N, \tag{7}$$

где $g > 0$ — максимальная емкость склада.

Учитывая представление (3), получаем

$$0 \leqslant M_j(k)U_j(k) \leqslant g - \eta_j(k). \tag{8}$$

В течение планового периода в N дней вводится $(N \times N)$-матрица M_N, строки которой имеют вид $M_k = (M_j(k), 0, \ldots, 0)$, $k = j, \ldots, N$. В каждой строке M_k подстрока нулей содержит $N - k$ нулей. Аналогично обозначим 0_N – нулевой N-мерный вектор и вектор G_N, элементы которого равны $g - \eta_j(k)$. Тогда (8) можно записать в векторной форме:

$$0_N \leqslant M_N U(N) \leqslant G_N. \tag{9}$$

Кроме того, логика торгового предприятия предполагает прямые ограничения на элементы вектора управления:

$$0 \leqslant u(k) \leqslant u_{\max}, \quad k = j, \ldots, N, \tag{10}$$

где $u_{\max}$ – константа, определяющая максимально возможный дневной объем оптовых поставок.

Когда $k = j = 1$, известен фактический остаток продаж $x_r(1)$. На основе этих начальных данных и уравнения (2) рассчитываются остатки запасов для будущих дней $k > j$, а затем, используя задачу линейного программирования (6)–(10), определяем стратегию заказов с первого по последний день. Когда $j > 1$, оптимизация проводится на основе нового фактического начального значения $x_r(j)$ для последующих $N - j$ дней. Таким образом, фазовая переменная $x(k)$ отражает фактический остаток запасов после каждого рабочего дня, а оптимальная стратегия ежедневных оптовых закупок реализуется на основе обратной связи по состоянию $x_r(j)$.

3. Численная реализация. Для проверки эффективности предложенного метода были взяты статистические данные торгового

предприятия, специализирующегося на продаже бананов в городе Сиань, за период с 1 октября 2024 года по 7 октября 2024 года (см. таблицу). В течение этого недельного планового периода необходимо рассчитать оптимальную стратегию пополнения запасов магазинов и сравнить результаты ее реализации с фактической прибылью. В колонке «Запасы» показаны начальные запасы бананов при открытии магазина, затем фиксируются объемы продаж, количество испорченного товара и объем заказов на следующий день. Значения всех этих величин указаны в кг. Последняя колонка указывает цену продажи за килограмм в юанях. Себестоимость бананов составляет 15 юаней за килограмм. Ежедневная стоимость хранения составляет 0,55 юаня за килограмм.

Таблица. Запасы и продажи бананов (01.10.2024 – 07.10.2024)

Дата	Запасы	Продажи	Испорчено	Заказы	Цена
01.10	98,57	55,00	11,34	60	46,6
02.10	92,23	19,26	9,50	50	46,6
03.10	113,47	49,66	10,00	80	46,6
04.10	133,81	85,81	13,00	70	21,54
05.10	105,00	28,68	8,00	50	45,9
06.10	118,32	67,80	11,00	50	45,9
07.10	89,52	27,52	12,50	50	45,9

Параметр оценки a – это среднедневная доля испорченных товаров и может быть вычислен по следующей формуле:

$$a(k) = \frac{x(k-1) - x(k) + u(k) - D(k)}{x(k-1) - D(k)}.$$

Для данных из таблицы $a(k) = [0{,}26; 0{,}13; 0{,}16; 0{,}27; 0{,}10; 0{,}22; 0{,}20]$.

В рамках данного примера фактический объем запасов составил $x_r(k) = [36{,}5; 63{,}9; 49{,}24; 31{,}43; 65{,}75; 42{,}95; 59{,}38]$. Кроме того, известны следующие параметры: максимальная емкость склада $g = 150$ кг; максимальный объем заказа $u_{max} = 90$ кг. С помощью нелинейной аппроксимации, основанной на статистических данных о ценах продажи и доле продаж, был найден параметр $\beta = 0{,}851$.

Далее была написана компьютерная программа для идентификации всех элементов модели и решения задачи линейного программирования (6)–(10). В результате численных экспериментов получена оптимальная стратегия заказов розничного торговца на неделю:

$$U(N) = \begin{pmatrix} 90 & 86{,}1 & 90 & 90 & 84{,}25 & 90{,}0 & 0 \end{pmatrix}^T.$$

4. Обсуждение результатов. Оптимальная стратегия на последний день равна нулю исключительно из-за временного интервала планирования. Было необходимо оптимизировать прибыль за неделю, что очевидно делает размещение заказов на последний день нецелесообразным, поскольку это увеличивает затраты на хранение и количество испорченного товара, а партия заказов в последний день вообще не будет реализована. Однако в реальной жизни торговый процесс непрерывен, поэтому для повышения адекватности модели при оптимизации прибыли в определенном временном интервале необходимо немного расширить период планирования и использовать промежуточные результаты (соответствующие заданному интервалу) в качестве окончательного оптимального плана. Это возможно, так как предложенный метод обладает итерационной природой.

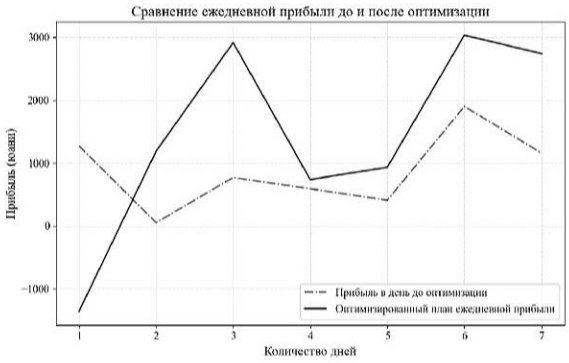


Рис. 1. Фактическая ежедневная прибыль магазина и прибыль при оптимальной скорости заказа

Рис. 1 демонстрирует графики прибыли до и после оптимизации. Согласно данным торгового предприятия, полученная прибыль за соответствующую неделю составила 6154,73 юаней. После применения оптимальной стратегии управления закупками товаров прибыль достигла 10181 юаня. Таким образом, прибыль увеличилась на $65,4\%$.

5. Заключение. В данном исследовании теория оптимального управления применена для оптимизации деятельности торгового

предприятия. Построена разностная система с управлением в виде ежедневных объемов оптовых закупок быстро портящихся товаров, а целью является максимизация прибыли с балансировкой избытка и недостатка товаров. Разработанный алгоритм способен обновлять и корректировать скорость заказов на основе фактического уровня запасов. Верификация на фактических данных китайской компании по продаже бананов показала, что оптимальная стратегия значительно превосходит существующие стратегии по прибыли. В заключении отметим направления дальнейшего развития модели, такие как учет товаров с различными уровнями свежести и решение многомерных задач оптимального управления, что имеет важное значение для практического применения.

Литература

1. Gabasov R., Kirillova F. M., Balashevich N. V. On the synthesis problem for optimal control systems // SIAM Journal on Control and Optimization. 2000. Vol. 39. No 4. P. 1008–1042.

2. Gabasov R., Kirillova F. M., Balashevich N. V. Synthesis of adaptive optimal controls for linear dynamic systems // Computational Optimal Control. 1994. P. 83–88.

3. Gharbi A., Kenne J. P., Kaddachi R. Dynamic optimal control and simulation for unreliable manufacturing systems under perishable product and shelf life variability // International journal of production economics. 2022. Vol. 247. P. 108417.

4. Polotski V., Gharbi A., Kenne J. P. Production control in manufacturing systems with perishable products under periodic demand // Journal of Manufacturing Systems. 2022. Vol. 63. P. 288–303.

УДК 51-77

Золотухина П. Ю.

Сравнение методов CPM и PERT планирования сроков выполнения проекта на примере организации выставки фотографий

Рекомендовано к публикации профессором Крылатовым А. Ю.

1. Введение. Методы PERT (Program Evaluation and Review Technique) и CPM (Critical Path Method) представляют собой инструменты, позволяющие оптимизировать процесс планирования, выявить критические задачи и минимизировать риски задержек. В данной статье рассматривается применение этих методов для планирования организации выставки фотографий.

2. Обзор методов. Рассматриваемые методы основаны на сетевом планировании, которое представляет проект в виде графа, состоящего из вершин (событий) и дуг (работ). Основная цель этих методов – определить критический путь, т. е. последовательность задач, которые имеют наименьший запас времени и напрямую влияют на общую продолжительность проекта [1].

2.1. Метод CPM (Critical Path Method). Этот метод помогает определить, какие задачи в проекте являются критическими, т. е. их задержка приведет к задержке всего проекта. Для этого проект разбивается на задачи с указанием их продолжительности и зависимостей. Затем рассчитываются ранние и поздние сроки начала и завершения каждой задачи, чтобы выявить резервы времени.

Ранние сроки (ES и EF) рассчитываются, начиная с первой задачи: ES – это самое раннее время начала задачи, а EF – это ES и продолжительность задачи. Поздние сроки (LS и LF) рассчитываются с конца проекта: LF – это самое позднее время завершения задачи, а LS – это LF без продолжительности этой задачи. Резерв времени (Slack) – это разница между поздним и ранним началом (или завершением) задачи, показывающая, на сколько можно задержать задачу без вреда для проекта. Задачи с нулевым резервом времени

Золотухина Полина Юрьевна – студент, Санкт-Петербургский государственный университет; e-mail: st087135@student.spbu.ru, тел.: +79811063132

образуют *критический путь*, т. е последовательность задач, которые нельзя задерживать.

2.2. Метод PERT (Program Evaluation and Review Technique). В основе данного метода лежит тот же самый принцип поиска критического пути, но оценка времени выполнения задачи в данном случае не задается условием задачи, а вычисляется исходя из трех оценок:

$$t_e = \frac{a + 4m + b}{6},$$

где a – оптимистичное время, m – наиболее вероятное время, b – пессимистичное время.

3. Постановка задачи. Исходные данные о задачах, которые необходимо выполнить для организации выставки, приведены в таблице 1.

Таблица 1. Исходные данные

Задача	**Описание**	**Зависимости**	a	m	b
A	Выбор концепции и темы	–	2	3	5
B	Поиск места проведения	A	5	7	10
C	Подбор фотографий	A	7	9	12
D	Подготовка экспозиции	C	8	10	15
E	Организация рекламной кампании	B, C	4	5	7
F	Проведение выставки	D, E	1	2	3
G	Демонтаж экспозиции	F	1	1	2

Эти данные будут использоваться в расчетах для обоих методов, но так как в методе CPM требуется только одна оценка времени, то в качестве нее будем использовать наиболее вероятное время m.

4. Применение методов для планирования работы над проектом. Рассмотрим на представленных в таблице 1 данных применение данных методов.

4.1. Определение критического пути с помощью CPM. В данном методе производится расчет сроков выполнения задач и резервов времени по следующим формулам. Результаты приведены в таблице 2.

Ранний срок начала:

$$ES = \max(EF_{\text{предшествующих задач}}).$$

Ранний срок окончания:

$$EF = ES + t_e.$$

Поздний срок окончания:

$$LF = \min(LS_{\text{последующих задач}}).$$

Поздний срок начала:

$$LS = LF - t_e.$$

Резерв времени:

$$\text{Slack} = LS - ES = LF - EF.$$

Таблица 2. Метод CPM

Задача	**ES**	**EF**	**LS**	**LF**	**Slack**
A	0	3	0	3	0
B	3	10	10	17	7
C	3	12	3	12	0
D	12	22	12	22	0
E	12	17	17	22	5
F	22	24	22	24	0
G	24	25	24	25	0

Критический путь – это последовательность задач с нулевым запасом времени, т. е. A – C – D – F – G.

Общая продолжительность проекта: 25 дней.

4.2. Определение критического пути с помощью метода PERT. Расчет времени выполнения задачи теперь производится по формуле

$$t_e = \frac{a + 4m + b}{6}.$$

Результаты вычислений для этого метода приведены в таблице 3.

Таблица 3. Метод PERT

Задача	t_e	**ES**	**EF**	**LS**	**LF**	**Slack**
A	3,17	0	3,17	0	3,17	0
B	7,17	3,17	10,34	10,50	17,67	7,33
C	9,17	3,17	12,34	3,17	12,34	0
D	10,5	12,34	22,84	12,34	22,84	0
E	5,17	12,34	17,51	17,67	22,84	5,33
F	2	22,84	24,84	22,84	24,84	0
G	1,17	24,84	26,01	24,84	26,01	0

Критический путь: A – C – D – F – G.
Общая продолжительность проекта: 26,01 дня.

5. Вывод. В обоих методах критический путь оказался одинаковым (A – C – D – F – G), что подтверждает важность этих задач для успешного завершения проекта. Однако PERT, благодаря учету неопределенности, дает более реалистичную оценку продолжительности проекта (26,01 дня против 25 дней в CPM).

Таким образом, выбор между CPM и PERT зависит от характера проекта. Если проект стабилен и предсказуем, то достаточно использования метода CPM. Если же проект связан с высокой неопределенностью, то метод PERT предоставляет более гибкий и точный инструмент для планирования и управления рисками [2, 3].

Литература

1. Антонова А. С., Аксенов К. А. Сравнительный анализ методов планирования субподрядных работ // Современные проблемы науки и образования. 2014. № 3. [Электронный ресурс]: URL:https://science-education.ru/ru/article/view?id=13388 (дата обращения: 20.03.2025).

2. Dr. Arati Shah PERT Vs. CPM: A cross review analysis // International Journal of Social Impact. 2021. No 6. P. 33–43.

3. Methaq Azeez Jouda, Mushtak A. K. Shiker. Comparing the effectiveness of PERT and CPM techniques in a house construction project: a case study // Journal Europeen des Systemes Automatises. 2024. No 57. P. 781–785.

УДК 519.677

Ибраева А. С.

Оценка эффективности модифицированного NSGA-II в задаче о назначениях: сравнение с точным методом и классическим NSGA-II

Рекомендовано к публикации доцентом Кумачевой С. Ш.

1. Введение. Одним из ключевых факторов результативности проекта является правильное распределение нагрузки лицом, принимающим решение (ЛПР). Проектный руководитель (или менеджер по персоналу) должен умело балансировать между возможностями сотрудников и их ресурсными ограничениями. Важно сохранять эффективность команды, при этом не делая ее численность критерием успеха.

Обобщенная проблема назначения (*Generalized Assignment Problem*, *GAP*) представляет собой комбинаторную задачу, в которой необходимо распределить n задач между m сотрудниками. Каждая задача должна быть закреплена за агентом, при этом требуемая ресурсная обеспеченность не должна превышать доступную. GAP является разновидностью задач о покрытии множества (Set Covering Problem, SCP) [1], а значит, относится к классу NP-трудных задач. Это обосновывает необходимость использования эвристических подходов, таких как генетические алгоритмы, для решения задач разумного размера.

В литературе предложено множество методов решения задачи GAP, включая генетический алгоритм (GA) [2], приближенные эвристики на основе задач [3], эвристики на основе множеств [4], поиск решений с учетом ограничений [5], метод ветвей и границ [6], алгоритм иерархической кластеризации [7] и другие.

Данная статья расширяет классическую однокритериальную постановку задачи GAP, вводя дополнительные критерии: расход ресурсов, загруженность сотрудников.

Такие изменения имеют практическую значимость. Например, при распределении задач в проекте ЛПР необходимо учитывать сразу несколько факторов, а не оптимизировать только один из них.

Ибраева Арина Сергеевна – студент, Санкт-Петербургский государственный университет; e-mail: st097356@student.spbu.ru, тел.: +7(917)434-76-54

Учитывая многокритериальный характер задачи, особое значение приобретает алгоритм NSGA-II – метод, разработанный для многокритериальной оптимизации с использованием Парето-сортировки. Эффективность NSGA-II по сравнению с классическим генетическим алгоритмом (GA) была доказана в ряде исследований [8].

Обзорные исследования, посвященные NSGA-II [9], показывают его широкое применение как в задачах GAP, так и в других комбинаторных задачах, что подтверждает актуальность модификации и исследования данного алгоритма.

2. Многокритериальная задача о назначении. Предположим, что $l = \{1, \ldots, m\}$ – множество кандидатов, $J = \{1, \ldots, n\}$ – множество задач. Определим c_{ij} как стоимость назначения кандидата i на задачу j, r_{ij} – ресурс, требуемый кандидату i для выполнения задачи j, b_i – доступный ресурс сотрудника i. Считаем, что x_{ij} – булева переменная, означающая назначение кандидата i на задачу j (1 – назначен, иначе 0).

$$f_{\text{efficiency}}(x) = \sum_{i \in I, j \in J} c_{ij} x_{ij} \to \max,$$

$$f_{\text{resource}}(x) = - \sum_{i \in I, j \in J} r_{ij} x_{ij} \to \max,$$

$$f_{\text{workload}}(x) = -\frac{n}{|Q|} \to \max,$$

где Q – множество назначенных специалистов.

$$\sum_{i \in I} x_{ij} = 1, \quad \forall j \in J, \qquad \sum_{i \in I, j \in J} r_{ij} x_{ij} \leqslant b_i, \quad \forall i \in I,$$

$$x_{ij} \in \{0, 1\}, \quad \forall i \in I, \forall j \in J.$$

3. Рассматриваемый алгоритм. В данной части используется описание эволюционного алгоритма NSGA-II, предложенного в [9, 10], где подробно рассмотрены основные принципы работы метода. За основу взята классическая схема, включающая генерацию популяции, селекцию, скрещивание и мутацию.

В рамках настоящей работы рассматривается модифицированная версия алгоритма, обозначенная как `enNSGA2`. Ключевым отличием данной модификации является использование евклидова расстояния в нормализованном пространстве критериев в качестве основного механизма селекции. После построения Парето-фронтов с

помощью процедуры *fast non-dominated sorting*, приоритет отдается тем решениям, которые находятся ближе всего к нормализованной идеальной точке — вектору, составленному из максимальных значений целевых функций по популяции. Это позволяет отдать предпочтение сбалансированным решениям, минимизирующим отклонение от одновременного достижения оптимума по всем критериям, и избегать экстремальных решений, оптимальных лишь по одной цели.

Формирование новой популяции происходит путём последовательного добавления особей, упорядоченных по фронтам и, внутри фронта, по евклидовой метрике, вплоть до достижения заданного размера.

Оператор скрещивания реализован в виде одноточечного кроссовера: потомки наследуют части хромосом от родителей по случайной точке разрыва.

Оператор мутации использует перемешивание индексов задач: случайным образом выбираются две задачи из хромосомы, и их исполнители обмениваются местами. Вероятность мутации задана на уровне 2%.

Псевдокод модифицированного алгоритма представлен ниже.

Require: N – размер популяции, P – начальная популяция
Ensure: A – множество недоминируемых решений
$t \leftarrow 0$
$P_t \leftarrow$ инициализация_популяции(N)
$A \leftarrow$ определить_недоминируемые(P_t)
while не выполнено условие остановки **do**
  $R_t \leftarrow P_t \cup Q_t$
  $F \leftarrow$ быстрая_Парето_сортировка(R_t)
  $P_{t+1} \leftarrow \emptyset$
  $i \leftarrow 1$
  while $|P_{t+1}| + |F_i| \leqslant N$ **do**
    $P_{t+1} \leftarrow P_{t+1} \cup F_i$
    $i \leftarrow i + 1$
  end while
  if $|P_{t+1}| < N$ **then**
    $F_i \leftarrow$ сортировка_по_евклидову_расстоянию(F_i)
    $P_{t+1} \leftarrow P_{t+1} \cup F_i[1 : (N - |P_{t+1}|)]$
  end if
  $Q_{t+1} \leftarrow$ селекция(P_{t+1}, N)
  $Q_{t+1} \leftarrow$ скрещивание(Q_{t+1})
  $Q_{t+1} \leftarrow$ мутация(Q_{t+1})
  $t \leftarrow t + 1$
  $A \leftarrow$ обновить_недоминируемые$(A \cup Q_t)$
end while
return A

---

4. Результаты. Алгоритм протестирован на шести датасетах из OR-Library [11], предназначенных для решения общей задачи о назначениях. Каждый датасет характеризуется количеством агентов и задач, матрицей эффективности, матрицей требуемых ресурсов, а также вектором доступных ресурсов. Рассматривались датасеты c515 (5 агентов и 15 задач), c520 (5 агентов и 20 задач), c832 (8 агентов и 32 задачи), c840 (8 агентов и 40 задач), c1050 (10 агентов и 50 задач) и c1060 (10 агентов и 60 задач). Для указанных наборов данных представлены точные решения, полученные методом ветвей и границ. В рамках настоящего исследования эффективность предложенной модификации алгоритма, обозначенной как enNSGA2, оценивалась путем сравнения с точным решением задачи однокритериальной оптимизации (с учетом затраченных ресурсов), а также с результатами классического алгоритма NSGA-II (baNSGA2). Для обеспечения сопоставимости оба эвристических алгоритма исследовались при идентичных наборах гиперпараметров, а именно, размер популяции – 50, 100 и 200 особей; вероятность скрещивания – 0,7; 0,8 и 0,9; вероятность мутации – 0,01; 0,02 и 0,05; максимальное число поколений – 50, 100 и 200. Каждая комбинация указанных гиперпараметров была оценена по результатам пяти независимых запусков алгоритма.

Алгоритм многокритериальной оптимизации должен демонстрировать устойчивость к изменению гиперпараметров, что позволяет исключить их точную настройку. Стабильность результатов при различных параметрах подтверждает их надежность и пригодность алгоритма для реальных задач с ограниченными ресурсами для подбора параметров. Многократные запуски дополнительно подтверждают, что полученные результаты являются закономерными.

Рассматриваемые критерии: максимизация эффективности назначения (f1), минимизация расхода ресурсов (f2), минимизация средней нагрузки на агента (f3).

Код был реализован на языке Python с помощью пакета DEAP, предназначенного для реализации генетических алгоритмов, а также ускорен применением кэширования функции оценки повторяющихся особей.

Код для реализации алгоритма доступен в репозитории [12].

Таблица 1. Сравнение алгоритмов на различных наборах данных

Набор данных	Алгоритм	f1 (оптимальность)	Отклонение f1 (%)	f2 (затраты)	Отклонение f2 (%)	f3 (загрузка агентов)	Отклонение f3 (%)	Время (сек)
c515	Точный	336,0	–	−164,0	–	−3,0	–	0,1
	baNSGA2	325,9	3,0	−273,8	67,0	−3,7	22,4	2,5
	enNSGA2	262,4	21,9	−150,8	8,1	−3,0	1,2	0,9
c520	Точный	434,0	–	−236,0	–	−4,0	–	0,1
	baNSGA2	420,6	3,1	−387,5	64,2	−4,9	23,5	1,1
	enNSGA2	292,7	32,6	−218,7	7,3	−4,0	0,9	0,7
c832	Точный	761,0	–	−362,0	–	−4,0	–	0,3
	baNSGA2	718,1	5,6	−634,5	75,3	−4,7	17,2	1,7
	enNSGA2	569,6	25,2	−313,0	13,5	−4,1	1,3	0,7
c840	Точный	942,0	–	−457,0	–	−5,0	–	0,8
	baNSGA2	892,3	5,2	−774,2	69,4	−5,5	11,5	1,9
	enNSGA2	708,3	24,8	−402,5	11,9	−5,0	0,7	0,7
c1050	Точный	1139,0	–	−611,0	–	−5,0	–	0,3
	baNSGA2	1022,0	10,3	−953,9	56,1	−5,6	12,9	2,0
	enNSGA2	698,0	38,7	−552,6	9,6	−5,1	1,0	0,8
c1060	Точный	1451,0	–	−698,0	–	−6,0	–	0,5
	baNSGA2	1344,3	7,4	−1165,0	66,9	−6,4	7,5	3,5
	enNSGA2	1094,0	24,6	−630,6	9,7	−6,0	0,4	0,8

На основании полученных результатов можно сделать следующие выводы относительно эффективности модифицированного алгоритма enNSGA2 в сравнении с точным методом решения задачи, а также классическим алгоритмом NSGA-II (baNSGA2).

1. Хотя метод ветвей и границ демонстрирует высокую эффективность на задачах ограниченного размера, его применение при масштабировании существенно ограничено. Это связано с экспоненциальным ростом вычислительной сложности при увеличении размерности пространства переменных, что подчёркивается в современных обзорах, например, в [13], а также в ранних исследованиях по задаче о назначениях [14]. В этих условиях

оправданным представляется переход к эвристическим методам, обладающим более стабильными характеристиками масштабируемости.

2. Рассматриваемые алгоритмы демонстрируют различное поведение в зависимости от критерия оптимизации. baNSGA2 достигает более высокой точности по критерию f_1, однако демонстрирует значительное отклонение по f_2. Для `c1060` baNSGA2 показал отклонение по f_2 на 66,94%, в то время как enNSGA2 – лишь 9,65%. Это свидетельствует о том, что baNSGA2 стремится к более точному назначению агентов, увеличивая расход ресурсов. enNSGA2, напротив, обеспечивает более сбалансированные результаты, демонстрируя лучшие показатели по f_2 и f_3. В частности, для `c520` значение f_3, полученное enNSGA2, отличается от точного решения всего на 0,91%, тогда как у baNSGA2 это отклонение составляет 23,52%.
3. Анализ полученных данных показывает, что с увеличением количества агентов и задач различия между алгоритмами становятся более явными. Разница в точности f_1 между baNSGA2 и enNSGA2 возрастает на более сложных задачах. Например, для `c515` отклонение enNSGA2 от точного решения составляет 21,90%, а для `c1060` уже 24,55%, что свидетельствует о тенденции к снижению точности решения с ростом размерности задачи.
4. Время выполнения enNSGA2 остается относительно стабильным, тогда как baNSGA2 значительно замедляется. Это подтверждает, что enNSGA2 лучше масштабируется на более сложные задачи.

5. Выводы. В статье предлагается новый подход к решению задачи о назначениях, основанный на применении алгоритма NSGAII.

Результаты проведенных экспериментов демонстрируют преимущества модифицированной версии алгоритма, заключающиеся в получении более сбалансированных решений по сравнению с классическим NSGA-II.

В классическом NSGA-II отбор решений происходит на основе их доминирования в многокритериальном пространстве. Однако такой подход может приводить к выбору экстремальных решений, не учитывающих баланс между критериями: если одно решение обладает

высокой эффективностью, но требует значительных ресурсных затрат, а другое решение, напротив, является менее эффективным, но при этом минимизирует потребление ресурсов, оба варианта могут быть одинаково приоритетными и попасть в число лучших решений.

В отличие от классического метода, предложенная модификация алгоритма обеспечивает учет баланса между критериями, избегая выбора экстремальных решений, что приводит к более равномерному покрытию множества допустимых решений в поисковом пространстве. Это позволяет повысить эффективность решения многокритериальной задачи назначений.

Литература

1. Sapkota N., Reilly C. H. Simulating realistic set covering problems with known optimal solutions // Computers & Industrial Engineering. 2011. Vol. 61. No 1. P. 39–47.

2. Chu P. C., Beasley J. E. A genetic algorithm for the generalised assignment problem // Computers and Operations Research. 1997. Vol. 24. No 1. P. 17–23.

3. Nauss R. M. Solving the generalized assignment problem: An optimizing and heuristic approach // INFORMS Journal on Computing. 2003. Vol. 15. No. 3. P. 249–266.

4. Cattrysse D. G., Salomon M., van Wassenhove L. N. A set partitioning heuristic for the generalized assignment problem // European Journal of Operational Research. 1994. Vol. 72. No. 2. P. 167–174.

5. Osman I. H. Heuristics for the generalized assignment problem: Simulated Annealing and Tabu Search approaches // OR Spektrum. 1995. Vol. 17. No. 3. P. 211–225.

6. Ринчинов Р. Ц., Парфенов А. П. Многокритериальная задача оптимального размещения производства в сети // Процессы управления и устойчивость. 2017. Т. 4. № 1. С. 667–671.

7. Лежнина Е. А., Хитров Г. М., Калинина Е. А., Кононов И. А. Рабочие группы: создание и управление // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2022. Т. 18. № 4. С. 608–620.

8. Zhang Sh., Guo H. et al. Multistage Assignment Optimization for Emergency Rescue Teams in the Disaster Chain // Knowledge-Based Systems. 2017. Vol. 137. P. 1–10.

9. Verma S., Pant M., Snasel V. A Comprehensive Review on NSGA-II for Multi-Objective Combinatorial Optimization Problems // IEEE Access. 2021. Vol. 9. P. 57757–57791.

10. Subtil R. F., Carrano E. G. et al. Using an Enhanced Integer NSGA-II for Solving the Multiobjective Generalized Assignment Problem // IEEE Congress on Evolutionary Computation. 2010. P. 1–7.

11. Generalized Assignment Problem (GAP) [Электронный ресурс] // URL: `https://people.brunel.ac.uk/~mastjjb/jeb/orlib/gapinfo.html` (дата обращения: 10.02.2023).

12. Код для реализации алгоритма [Электронный ресурс] // URL: `https://github.com/illneverwakeup/assignment_problem/blob/3024cc47474a836f2d05ea82762aa30450974fa7/enNSGA2.py` (дата обращения: 18.10.2023).

13. Huang L. et al. Branch and bound in mixed integer linear programming problems: a survey of techniques and trends // arXiv preprint, 2021. https://arxiv.org/abs/2111.06257v1

14. Ross G. T., Soland R. M. A branch and bound algorithm for the generalized assignment problem // Mathematical Programming. 1975. Vol. 8. P. 91–103.

УДК 51-77

Карабаев Е. Е.

Анализ выпуска продуктов птицеводства с помощью динамических систем

Рекомендовано к публикации профессором Крылатовым А. Ю.

Введение. Динамические системы имеют различные применения в экономике и общественных науках. В основном они могут применяться при анализе демографических процессов, как например в [1], или в анализе межотраслевых связей в течение времени, как в [2]. Анализ же отдельных рынков товаров и особенностей их производства с помощью динамических систем редко встречается в современной литературе.

Куриные яйца являются одним из основных продуктов питания и важным источником белка для организма. Они потребляются как отдельный продукт, так и в составе других товаров. Как и другие продукты птицеводства, яйца получаются от производительных птиц. Из-за внешних факторов, эпидемий, природных катастроф и т. д., численность продуктивных птиц может резко сократиться. Это может привести к дефициту яиц.

В этом случае необходимо выяснить, сможет ли численность курнесушек восстановиться до прежних значений и за какой срок. В данной работе предлагаются способы математического моделирования численности производительных птиц с учётом особенностей их разведения.

Описание процесса производства. Длительность жизни кур ограничена, а продуктивность (яйценосность) снижается после определённого возраста. Чтобы избежать падения производства яиц, несушки по достижении определённого возраста подлежат отбраковке.

Для поддержания требуемого уровня производства необходимо возмещать ранее выбракованных птиц. С этой целью часть яиц идёт на инкубацию – процесс искусственного или естественного разведения и выращивания птиц.

Карабаев Егор Евгеньевич – магистрант, Санкт-Петербургский государственный университет; e-mail: eekarabaev@mail.ru, тел.: +7(915)082-80-62

Яйца отправляются в специальные хранилища – инкубаторы, где поддерживаются оптимальные условия для вылупления цыплят. Затем из вылупившихся цыплят отделяют птиц женского пола, примерно половину всех птенцов.

Птенец вылупляется через 3 недели, а продуктивного возраста достигает в 22 недели. Таким образом, от момента выведения яйца до куры-несушки проходит 25 недель.

Математические модели описания процесса. В птицеводстве используются различные математические методы. Например, модели производительности кур-несушек описаны в работе [3]. Численность и воспроизводство популяций описываются моделями популяционной динамики.

У птиц между рождением и достижением продуктивного возраста проходит большой промежуток времени. В этом случае можно описать процесс воспроизводства с помощью динамических систем с запаздыванием, которые подробно описаны в [4].

Опишем воспроизводство кур-несушек с помощью динамической системы вида:

$$\dot{n} = an(t) + bn(t - \tau), \tag{1}$$

где $\dot{n} = \frac{dn}{dt}$, первое слагаемое это количество выбывших птиц, второе – поступивших из инкубации. Эти величины должны быть равны, так как необходимо поддерживать постоянную численность кур-несушек.

$$\dot{n} = -(m + r)n(t) + \frac{1}{2}hp \cdot inc \cdot n(t - \tau), \tag{2}$$

где коэффициенты при означают следующее: m – смертность кур, примерно 0,2% в неделю; r – выбраковка кур, примерно 1,3% в неделю; h – выводимость, выводится из яйца и выживает до возраста куры-несушки около 95% цыплят; $\frac{1}{2}$ – доля кур женского пола; p – яйценосность куры, кура яичной породы сносит в среднем 4 яйца в неделю; inc – доля яиц на инкубацию, примерно 0,8%;

Таким образом, система имеет следующий вид:

$$\dot{n} = -0{,}015n(t) + 0{,}015n(t - \tau).$$

Отметим, что все коэффициенты модели, кроме доли яиц на инку-

бацию inc, постоянны. Доля на инкубацию может меняться в зависимости от решений производителей.

Устойчивость модели. Проверим, будет ли численность кур-несушек n(t) стремиться к прежнему значению при изменении начальной численности и заданных параметрах выбраковки и воспроизводства.

Для этого следует исследовать нашу систему на асимптотическую устойчивость по Ляпунову. Пусть она задана следующим образом:

$$\begin{cases} \dot{n} = an(t) + bn(t-\tau), \ t > 0, \\ n(0) = N, \end{cases} \tag{3}$$

где N – начальная численность.

Определение. Система (3) называется асимптотически устойчивой по Ляпунову, если:

$$\forall \varepsilon > 0 \ \exists \delta = \delta(\varepsilon) > 0 : \forall N_0 : \rho(N, N_0) < \delta \Rightarrow \lim_{t \to \infty} \rho(n(t), n_0(t)) = 0.$$

Чтобы система (3) была асимптотически устойчива, все действительные части корней характеристического уравнения должны иметь отрицательные действительные части. Проверить это можно с помощью теоремы Хейза:

Теорема [4]. *Все корни характеристического уравнения имеют отрицательные действительные части тогда и только тогда, когда выполняются следующие неравенства:*

1. $a\tau < 1$.
2. $a + b < 0$.
3. $b\tau + \sqrt{(a\tau)^2 + \mu^2} > 0$, *где* $\mu = a\tau \tan \mu$.

В нашем случае $a + b = 0$, что противоречит условию 2 теоремы. Это говорит об асимптотической неустойчивости системы (1). Значит, если количество кур-несушек изменится под воздействием внешних факторов, оно не вернётся к прежнему. Чтобы восстановить прежнюю численность, необходимо изменить параметры системы.

Учёт инкубационных мощностей. Как было упомянуто ранее, единственный параметр, который могут изменить производители – это доля яиц, отправляемых на инкубацию. Восстановить численность можно с помощью увеличения доли яиц, отправляемых в инкубаторы.

Однако инкубаторы не могут резко увеличить количество принимаемых яиц. В краткосрочном периоде инкубационные мощности остаются неизменными. Представим максимальное количество кур, которое может выпускать инкубатор, в виде константы Capacity $=$ const > 0.

Если прежняя численность была равна N, то инкубаторы были рассчитаны на выращивание Capacity $= \frac{1}{2}hp \cdot inc \cdot N = 0,015N$ кур в неделю. При сокращении начальной численности до αN, $\alpha < 1$, чтобы заполнить инкубатор, новая доля отложенных яиц должна увеличиться до $\frac{inc}{\alpha} > inc$. Тогда

$$\frac{1}{2}hp\frac{inc}{\alpha}\alpha N \;=\; 0{,}015N \;=\; \text{Capacity}.$$

Таким образом, можно представить систему (2) в виде:

$$\dot{n} = \text{Capacity} - (m + r)n, \quad n(0) = N.$$

Проверим уравнение (2) на асимптотическую устойчивость. Корень характеристического уравнения

$$\lambda = -(m + r).$$

Так как $m+r > 0$, то $\lambda < 0$. Следовательно, система асимптотически устойчива. Это означает, что при изменении начального поголовья кур-несушек и максимальной загрузке инкубационных мощностей, их количество будет стремиться к прежнему уровню с течением времени.

Найдём функцию численности:

$$\int \dot{n}\, dt = \int (\text{Capacity} - (m + r)n)dt = \frac{\text{Capacity}}{m + r} - \frac{C}{e^{(m+r)t}}.$$

В зависимости от начального решения $n(0)$ константа C будет принимать разные значения. Как показано ранее, инкубационные мощности были рассчитаны на поддержание постоянной численности N и компенсацию выбывших кур, т. е.

$$\frac{\text{Capacity}}{m + r} = \frac{(m + r)N}{m + r} = N.$$

Если же начальная численность $n(0)$ изменилась до αN, то при $t = 0$ получим

$$n(0) = N + \frac{C}{e^{(m+r)0}} = \alpha N,$$

$$N + C = \alpha N,$$
$$C = (\alpha - 1)N.$$

Таким образом, численность кур-несушек при максимальной нагрузке инкубационных мощностей, обеспечивающих компенсацию выбывших кур, можно представить следущим образом:

$$n(t) = N - \frac{(1-\alpha)N}{e^{(m+r)t}}.$$

При $t = 0$ имеем $n(0) = \alpha N$. При $t \to \infty$ – $n(t) \to N$.

Время восстановления численности. Функция численности стремится к прежней N и никогда её не достигнет. На практике же поголовье на птицефабриках немного выше необходимого (примерно на 5–10%). Выясним, когда численность достигнет 90% от N.

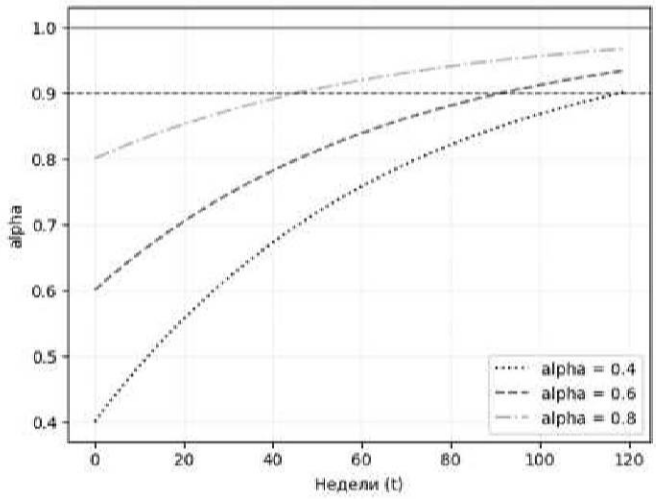


Рис. 1. Восстановление численности кур-несушек при изменении численности до 40%, 60% и 80% от прежней

На рис. 1 представлены графики численности кур-несушек для $\alpha = 0{,}4$; 0,6 и 0,8. Рассмотрим, в каких точках графики пересекали прямую необходимой численности.

Таблица. Количество недель до восстановления численности

alpha	Неделя
0,8	46
0,6	82
0,4	119

Как видно из таблицы, два значения превышают количество недель в году. При таком долгом периоде возможно увеличение инкубационных мощностей и уменьшение срока восстановления. Это делает расчёты нерелевантными.

Рассмотрим, в какой мере может измениться начальная численность, чтобы поголовье восстановилось до необходимого не более, чем за год (52 недели). Сначала найдём t

$$N - \frac{(1-\alpha)N}{e^{(m+r)t}} = 0{,}9N, \quad 1 - \frac{(1-\alpha)}{e^{(m+r)t}} = 0{,}9,$$

$$e^{(m+r)t} = 10(1-\alpha), \quad t = \frac{1}{m+r}\ln(10-10\alpha).$$

Подставим показатели смертности и выбраковки, а также ограничим время 52-мя неделями.

$$\frac{1}{0{,}015}\ln(10-10\alpha) \leqslant 52, \quad \ln(10-10\alpha) \leqslant 0{,}78,$$

$$10-10\alpha \leqslant e^{0{,}78}, \quad \alpha \geqslant \frac{10-e^{0{,}78}}{10}, \quad \alpha \geqslant 0{,}78.$$

Таким образом, если численность кур-несушек упала не менее чем до 78% от начальной, то поголовье восстановится в пределах года.

Заключение. В работе рассмотрена возможность представления производства яиц и воспроизводства кур-несушек в виде динамических систем, в том числе с запаздыванием. Полученные системы проверены на наличие асимптотической устойчивости. Показано, что максимальная загрузка инкубационных мощностей позволяет восстановить численность при потере части кур-несушек от внешних факторов. При падении до 78% от первоначальной численности поголовье может восстановиться до необходимого уровня за срок до года.

Литература

1. Балыкина Ю. Е., Захаров В. В. Интегральная модель притока и оттока и ее приложения // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2024. Т. 20. № 2. С. 121–135.

2. Смирнов Н. В., Смирнова Т. Е., Смирнова М. А., Смирнов М. Н. Разностные динамические модели межотраслевого баланса // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2023. Т. 19. № 1. С. 51–64.

3. Yang N., Wu C., McMillan I. A. N. New mathematical model of poultry egg production // Poultry Science. 1989. Vol. 68. No 4. P. 476–481.

4. Полянин А. Д., Сорокин В. Г., Журов А. И. Дифференциальные уравнения с запаздыванием: Свойства, методы, решения и модели. М.: Изд-во «ИПМех РАН», 2022. 464 с.

УДК 519.683.8

Кремко В. В.

Анализ и оценка эффективности методов машинного обучения в прогнозировании цен на продукты питания

Рекомендовано к публикации профессором Крылатовым А. Ю.

1. Введение. Цены на продукты питания оказывают значительное влияние на экономику, так как определяют уровень жизни населения и общее состояние рынка. Прогнозирование цен особенно важно в условиях экономической нестабильности, изменения климата и роста объёмов данных, что требует применения современных методов анализа и моделирования.

Точность прогнозирования цен на продукты имеет большое значение для различных сфер, включая розничную торговлю, логистику и управление запасами. На процесс влияет множество факторов: сезонность, изменения спроса и предложения, мировые экономические тренды, стоимость сырья, климат и другие внешние условия.

Методы машинного обучения предоставляют инструменты для анализа данных и построения прогнозных моделей. Они позволяют эффективно обрабатывать большие объёмы информации, выявлять скрытые закономерности и формировать точные предсказания. Преимущество методов машинного обучения заключается в их способности учитывать нелинейные зависимости, сезонные колебания и внешние макроэкономические факторы.

Целью исследования является сравнительный анализ различных алгоритмов машинного обучения для прогнозирования цен на продукты питания с оценкой их точности и применимости на практике.

2. Обзор данных. В работе использовались три набора данных, охватывающие период с 2008 по 2020 год. Данные содержат информацию о ценах на продукты питания в России, макроэкономических показателях и климатических условиях.

Кремко Вера Валерьевна – студент, Санкт-Петербургский государственный университет; e-mail: st097354@student.spbu.ru, тел.: +7(964)146-23-15

Данные о ценах на продукты питания были получены из базы Всемирной продовольственной программы и охватывают 98 регионов России. Датасет содержит 11006 строк с ежемесячными значениями цен на основные продовольственные товары. В данных указаны дата фиксации цены (15-е число каждого месяца), регион и субъект Российской Федерации, где была зафиксирована цена, наименование продукта, категория (злаки, молочные продукты, овощи и фрукты), единица измерения (килограммы, литры), а также цена в рублях.

Макроэкономические показатели включают информацию о ключевой ставке и уровне инфляции, полученную из открытых данных Центрального банка России. Эти данные имеют ежемесячную периодичность и охватывают тот же временной интервал, что и данные о ценах на продукты. Инфляция показывает, как быстро растут цены на товары и услуги, что влияет на уровень спроса и покупательскую способность.

Климатические данные были получены из базы Climate Hazards Group InfraRed Precipitation (CHIRPS). Они содержат информацию о среднемесячном количестве осадков по регионам России. Данные получены на основе спутниковых наблюдений и агрегированы по административным единицам. Климатические факторы оказывают значительное влияние на урожайность сельскохозяйственных культур, что, в свою очередь, сказывается на объёмах предложения и уровне цен на продукты питания.

3. Подготовка данных. Перед применением методов машинного обучения была выполнена предобработка данных, включающая:

1. Очистку данных. На первом этапе очистки был проведён тщательный анализ пропущенных значений. В случае обнаружения отсутствующих данных, они были заменены на статистически обоснованные показатели, такие как среднее или медиану, в зависимости от особенностей конкретных переменных. Для критически важных пропусков, которые могли существенно повлиять на результаты анализа, было принято решение удалить соответствующие строки. Это позволило минимизировать искажения, вызванные неполными записями. Несмотря на это, основной набор данных оказался достаточно чистым и не содержал значительных пропусков. Однако были выявлены дубликаты. В результате была удалена 21 строка с повторяющейся информацией.

Далее была проведена очистка выбросов с использованием метода межквартильного размаха Interquartile Range (IQR). Этот метод позволяет эффективно выявлять и устранять аномальные значения, которые могут искажать модель. Были определены нижняя и верхняя границы для каждого признака, и значения, выходящие за эти пределы, были удалены.

2. Приведение дат к единому формату. Все даты в наборе данных были преобразованы в стандартный формат datetime, что обеспечило корректную работу с временными рядами и позволило эффективно работать с признаками и анализировать сезонные колебания и другие временные эффекты, влияющие на ценообразование.

3. Объединение данных. В рамках работы использовалось три основных набора данных: данные о ценах на продукты питания, макроэкономические показатели и климатические факторы. Для обеспечения целостности анализа было проведено объединение информации в единый датасет. При объединении учитывались все возможные несоответствия в форматах и типах данных, что гарантировало согласованность информации для дальнейшего анализа.

4. Методы прогнозирования. В исследовании были использованы три модели машинного обучения.

4.1. SARIMA. В ходе работы для построения прогноза цен на продукты выбрана модель SARIMA (Seasonal AutoRegressive Integrated Moving Average). Эта модель представляет собой расширение классической модели ARIMA [1], ее использование позволяет более точно моделировать поведение временных рядов, учитывая сезонность данных.

Модель ARIMA(p, d, q) определяется для временных рядов y_t как порядок авторегрессии p, порядок дифференцирования d и порядок скользящего среднего q. Общий вид функции можно вывести следующим образом [2]:

$$(1 - \sum_{i=1}^{p} \alpha_i L^i)(1 - L^d) y_t = (1 - \sum_{j=1}^{q} \theta_j L^j)\varepsilon_t,$$

где L – оператор сдвига (лаговый оператор), α_i – параметры авторегрессионной части модели порядка L^i, θ_i – параметры части скользящего среднего порядка L^i, ε_t – ошибка в момент времени t, которая предполагается нормально распределённой с нулевым средним и постоянной дисперсией.

В модели SARIMA добавляется параметр s – длина сезонного цикла [3], или количество наблюдений в течение одного сезона.

4.2. XGBoost (Extreme Gradient Boosting). Это метод градиентного бустинга на деревьях решений, который показывает высокую точность на сложных данных [4].

Метод использует регуляризацию для предотвращения переобучения. Для настройки гиперпараметров использовался метод случайного поиска (Random Search). Для минимизации ошибки в процессе обучения используется функция потерь

$$L(y, \widehat{y}) = \sum_{i=1}^{n} (y_i - \widehat{y}_i)^2 + \lambda \|w\|^2.$$

4.3. LSTM (Long Short-Term Memory). Данная модель – разновидность рекуррентных нейронных сетей, которая хорошо подходит для задач прогнозирования временных рядов с учётом длинных временных зависимостей и сезонных закономерностей. В отличие от классических статистических методов, LSTM позволяет эффективно захватывать нелинейные зависимости и скрытые закономерности в данных [5].

Временной шаг составлял 12 месяцев, что позволило учесть сезонные колебания в данных. Данные были разбиты на обучающую и тестовую выборки в соотношении 80:20.

Архитектура модели включала:

- два слоя LSTM с количеством нейронов 50 в каждом слое;
- слои Dropout с коэффициентом 0,2 для предотвращения переобучения;
- полносвязный слой Dense для формирования выходного значения.

В качестве функции потерь использовалась среднеквадратичная ошибка (MSE), а оптимизация выполнялась с помощью алгоритма Adam. Обучение модели проводилось в течение 20 эпох с размером батча 16.

5. Результаты. Для оценки качества модели были использованы две основные метрики [6]:

- RMSE (Root Mean Squared Error) – среднеквадратичная ошибка, которая рассчитывается как

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2},$$

  где y_i – истинные значения, $\widehat{y}_i$ – предсказанные значения, n – количество данных.

- MAPE (Mean Absolute Percentage Error) – средняя абсолютная процентная ошибка. Показывает средний процент отклонения прогнозируемых значений от фактических. Значение ошибки рассчитывается как

$$\text{MAPE} = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{y_t - \widehat{y}_t}{y_t}\right| \times 100,$$

  где n – количество наблюдений, y_t – фактическое значение временного ряда в момент времени t, $\widehat{y}_t$ – прогнозируемое значение временного ряда в момент времени t, $\left|\frac{y_t - \widehat{y}_t}{y_t}\right|$ – абсолютное относительное отклонение прогноза от фактического значения.

Результаты сравнения моделей на тестовых данных приведены в таблице.

Таблица. Сравнение методов машинного обучения

Метод машинного обучения	**RMSE**	**MAPE (%)**
SARIMA	16,6	18,3
XGBoost	16,6	19,2
LSTM	13,1	12,3

6. Выводы. На основании представленных результатов сравнения моделей машинного обучения при прогнозировании цен на продукты можно сделать следующие выводы.

Модель SARIMA показала наиболее сбалансированные результаты среди всех тестируемых методов. RMSE для SARIMA составил 16,6, что указывает на приемлемую точность прогноза, а MAPE – 18,3%, что свидетельствует об относительно невысоком проценте ошибок при прогнозировании цен.

Модель XGBoost продемонстрировала аналогичные показатели с SARIMA: RMSE также составил 16,6, но MAPE был немного выше – 19,2%. Это свидетельствует о том, что XGBoost не продемонстрировал значительного улучшения по сравнению с SARIMA при прогнозировании цен на продукты.

Модель LSTM продемонстрировала лучшие результаты среди всех тестируемых методов. RMSE для LSTM составил 13,1, что является наименьшим значением среди всех моделей, а MAPE – 12,3%, что также указывает на лучшую точность прогноза. Эти результаты подтверждают высокую эффективность LSTM для решения задачи прогнозирования, особенно при работе с временными рядами, где глубокие нейронные сети способны учитывать сложные нелинейные зависимости.

Таким образом, на основе представленных данных можно сделать вывод, что модель LSTM является наиболее точной для прогнозирования цен на продукты, в то время как методы SARIMA и XGBoost показывают схожие, но менее эффективные результаты.

Результаты исследования подтверждают эффективность применения алгоритмов машинного обучения для прогнозирования цен на продукты питания.

Литература

1. Ци Д., Буре В. М. Исследование методов прогнозирования временных рядов для предсказания качества воздуха: объяснительный сравнительный анализ // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2024. Т. 20, № 2. С. 206–219.

2. MacLachlan M. J., Chelius C. A., Short G. Time-series methods for forecasting and modeling uncertainty in the food price outlook // USDA, Economic Research Service. 2022. P. 4–17.

3. Kozuch A., Cywicka D., Adamowicz K. Comparison of artificial neural network and time series models for timber price forecasting // Forests. 2023. Vol. 14. P. 2–9.

4. Brownlee J. XGBoost with Python: gradient boosted trees with XGBoost and scikit-learn. San Francisco: Machine Learning Mastery, 2024. 111 p.

5. Korstanje J. Advanced forecasting with Python: with state-of-the-art-models including LSTMs, Facebook's Prophet, and Amazon's DeepAR. New York: Apress, 2021. 296 p.

6. Mitchell T. Machine learning. New York: McGraw Hill, 1997. 414 p.

УДК 51-77

Мова И. А.

Анализ потребительских и ипотечных кредитов по регионам России: пространственно-эконометрический подход

Рекомендовано к публикации доцентом Панкратовой Я. Б.

1. Введение. В настоящее время кредитование является одной из самых востребованных финансовых услуг среди населения. Влияние кредитования на экономику оценивается неоднозначно. С одной стороны, активное использование кредитов может стимулировать экономическое развитие за счёт оживления рынка недвижимости и повышения потребительского спроса, а также способствовать повышению качества жизни. С другой стороны, чрезмерная закредитованность населения может негативно влиять на экономику, провоцировать увеличение риска дефолтов и банковских кризисов.

В данной работе исследуется, существует ли зависимость по объёму и количеству потребительских и ипотечных кредитов между регионами. А также строятся пространственные модели для данных, имеющих пространственную зависимость по регионам.

Статические данные по субъектам Российской Федерации для анализа были получены с сайта Банка России [1] и Единой межведомственной информационной системы социального обеспечения [2].

2. Описание данных. В данной работе изучаются следующие данные:

- объём ипотечных кредитов (млн руб.);
- количество ипотечных кредитов (ед.);
- объём потребительских кредитов (млн руб.);
- количество потребительских кредитов (ед.).

Для построения пространственной модели используются данные:

Мова Ирина Александровна – магистрант, Санкт-Петербургский государственный университет; e-mail: st133232@student.spbu.ru, тел.: +7(905)814-41-42

- численность безработных в возрасте 15 лет и старше по субъектам Российской Федерации (чел.);
- численность постоянного населения по субъектам Российской Федерации (чел.);
- ожидаемая продолжительность жизни (лет);
- уровень бедности (доля численности населения с денежными доходами ниже величины прожиточного минимума в процентах к общей численности населения) (%);
- среднедушевые денежные доходы населения по субъектам Российской Федерации (руб./месяц);
- обеспеченность больничными койками на 10000 человек (шт.).

3. Описание математической модели. Необходимым компонентом для проверки наличия пространственной зависимости и построения пространственно-экономической модели является матрица пространственных весов.

Пространственные веса – это числа, отражающие расстояние некоторого рода между целевым пространственным объектом и любым другим в пределах указанной окрестности [3]. Двоичная матрица смежности – простейшая матрица пространственных весов. Два региона являются смежными или соседними, если они имеют общую границу. В этом случае пространственный вес, записывающийся в матрицу пространственных весов, равен единице, в противном случае пространственный вес равен нулю [4, 5].

Для проверки наличия пространственной зависимости одного региона на другой применяется индекс Морана [5]. Он оценивает статистическую взаимосвязь между значением показателя в данной территориальной единице и значениями исследуемого показателя в соседних территориальных единицах. Нулевой гипотезой этого теста является отсутствие пространственной автокорреляции. Альтернативная гипотеза – автокорреляция присутствует, т. е. хотя бы один из коэффициентов пространственной корреляции не равен нулю, что указывает на наличие взаимосвязи между значениями показателя в соседних территориальных единицах.

Дополнительно рассчитывается ожидаемое значение индекса Морана, которое используется для оценки его статистической значимости. Значимость рассчитанного значения индекса Морана определяется путем сравнения его с ожидаемым значением и стандартным отклонением. Для этой оценки применяется Z-тест Фишера, с помощью которого вычисляется P-значение – вероятность получить для данной вероятностной модели распределения значений случайной величины такое же или более экстремальное значение статистики, по сравнению с наблюдаемым, при условии, что нулевая гипотеза верна.

Уровень значимости – это пороговое значение, которое определяет, насколько вероятно, что наблюдаемый результат мог бы возникнуть случайно, если нулевая гипотеза верна. Сравнивая полученное P-значение с этим уровнем, мы принимаем решение об отклонении или принятии нулевой гипотезы. Если $p < 0,05$, то нулевая гипотеза отвергается. Если $p \geqslant 0,05$, нет оснований для отказа от нулевой гипотезы.

Анализ данных показал наличие пространственной зависимости для количества потребительских кредитов (таблица 1) и отсутствие зависимости для объёма потребительских кредитов.

Таблица 1. Результаты теста Морана для количества потребительских кредитов

Дата	Тест Морана	Ожидаемое значение	Z-тест	P-значение
01.01.2023	0,1242	−0,0133	1,9442	0,0259
01.07.2023	0,1108	−0,0133	1,7662	0,0387
01.01.2024	0,1979	−0,0133	2,9445	0,0016
01.07.2024	0,1302	−0,0133	2,0260	0,0214

Аналогично, существует пространственная зависимость для количества ипотечных кредитов (таблица 2) и отсутствие зависимости для объёма ипотечных кредитов.

Таблица 2. Результаты теста Морана для количества ипотечных кредитов

Ujl	Тест Морана	Ожидаемое значение	Z-тест	P-значение
2010	0,2262	−0,0128	3,4866	0,0002
2011	0,2070	−0,0128	3,2200	0,0006
2012	0,1995	−0,0128	3,1030	0,0010
2013	0,1876	−0,0128	2,9409	0,0016
2014	0,2076	−0,0128	3,2146	0,0007
2015	0,2109	−0,0128	3,2573	0,0006
2016	0,2070	−0,0128	3,2200	0,0006
2017	0,1995	−0,0128	3,1030	0,0010
2018	0,2262	−0,0128	3,4866	0,0002
2019	0,2070	−0,0128	3,2200	0,0006
2020	0,1995	−0,0128	3,1030	0,0010
2021	0,1876	−0,0128	2,9409	0,0016
2022	0,2076	−0,0128	3,2146	0,0007
2023	0,2109	−0,0128	3,2573	0,0006

Рассмотрим четыре похода к построению модели с пространственной зависимостью [6, 7]:

1. Модель с пространственным лагом (SAR). Эта модель включает пространственные лаги зависимой переменной, что позволяет учитывать влияние значений зависимой переменной в соседних регионах:

$$y_{it} = \rho \sum_{j=1}^{N} W_{ij} y_{jt} + X_{it}\beta + \mu_i + \sum_{k=1}^{m} \gamma_k d_k + \varepsilon_{it},$$

где ρ – пространственный коэффициент корреляции,W_{ij} – элементы взвешивающей матрицы, X_{it} – матрица объясняющих переменных, β – вектор оцениваемых коэффициентов при факторах, μ_i – индивидуальные фиксированные эффекты, γ_k – временные эффекты, ε_{it} – ошибки, i – номер региона, N – объем выборки по регионам, t – время, для потребительского кредита – полугодие, а для ипотечного – год.

2. Модель Дарбина (SDM). В этой модели, помимо пространственного лага, учитываются пространственно-взвешенные значения объясняющих переменных:

$$y_{it} = \rho \sum_{j=1}^{N} W_{ij} y_{jt} + X_{it}\beta + \sum_{j=1}^{N} W_{ij} X_{jt}\theta + \mu_i + \sum_{k=1}^{m} \gamma_k d_k + \varepsilon_{it}.$$

3. Модель с пространственной структурой остатков (SEM):

$$y_{it} = X_{it}\beta + \mu_i + \sum_{k=1}^{m} \gamma_k d_k + \varepsilon_{it}, \quad \varepsilon_{it} = \lambda \sum_{j=1}^{N} W_{ij}\nu_{jt} + \nu_{it}.$$

4. Модель с пространственным лагом и пространственной структурой в ошибках (SAC):

$$y_{it} = \rho \sum_{j=1}^{N} W_{ij} y_{jt} + X_{it}\beta + \mu_i + \sum_{k=1}^{m} \gamma_k d_k + \varepsilon_{it}, \quad \varepsilon_{it} = \lambda \sum_{j=1}^{N} W_{ij}\nu_{jt} + \nu_{it}.$$

Лучшими моделями для прогнозирования количества кредитов оказались SAR, SDM и SAC. Результаты этих моделей представлены в таблицах 3, 4, 5 соответственно, в скобках приведены значения стандартной ошибки, **, *** – значимость на уровнях в 0,05; 0,01 % соответственно:

- в модели с пространственным лагом коэффициент пространственной зависимости ρ имеет небольшое значение, но является значимым. Значимым фактором является численность населения региона.

Таблица 3. Результаты оценки моделеи с пространственным лагом

Переменные	SAR
Численность региона	$4{,}0833 \cdot 10^{-5}$ $(8{,}3718 \cdot 10^{-6})$***
ρ	0,031398 ***
σ_e^2	0,0026834 ***
Всего наблюдений	312
Количество регионов	78

- в модели Дарбина также коэффициент пространственной зависимости ρ имеет небольшое значение и является значимым. Значащими факторами являются среднедушевой доход и среднедушевой доход в предыдущий период.

Таблица 4. Результаты оценки модели Дарбина

Переменные	SDM
Среднедушевой доход	$5{,}3148{\cdot}10^{-3}$ $(7{,}9124{\cdot}10^{-4})$***
Среднедушевой доход в предыдущий период	$-3{,}9175 \cdot 10^{-4}$ $(6{,}6815{\cdot}10^{-5})$***
ρ	0,03485 ***
σ_e^2	0,0024864 ***
Всего наблюдений	312
Количество регионов	78

- в модели с пространственным лагом ρ и пространственной структурой в ошибках λ коэффициент пространственной зависимости является отрицательным и малым, но значимым. Коэффициент пространственной зависимости для ошибок имеет небольшое значение и является значимым. Значимыми факторами являются численность населения региона, среднедушевой денежный доход, и их значения на предыдущем периоде.

Таблица 5. Результаты оценки модели с пространственным лагом и пространственной структурой в ошибках

Переменные	SAC
Численность региона	$4{,}2987{\cdot}10^{-5}$ $(7{,}8393{\cdot}10^{-6})$***
Среднедушевой доход	$5{,}1489{\cdot}10^{-3}$ $(7{,}3461{\cdot}10^{-4})$***
Численность региона в предыдущий период	$3{,}4663{\cdot}10^{-6}$ $(1{,}4407{\cdot}10^{-6})$**
Среднедушевой доход в предыдущий период	$-5{,}8543 \cdot 10^{-4}$ $(9{,}8657{\cdot}10^{-5})$***
ρ	−0,0075084
λ	0,036999 ***
σ_e^2	0,0015058 ***
Всего наблюдений	312
Количество регионов	78

Аналогичные модели были получены для количества ипотечных кредитов. Каждая из этих моделей может быть использована для прогнозирования потребительских и ипотечных кредитов на следующий период для интересующего региона.

4. Заключение. В данном исследовании выявлено существование пространственной зависимости для количества потребительских и ипотечных кредитов по регионам Российской Федерации, а также построены пространственные модели, наилучшим образом описывающие данные.

Литература

1. Центральный банк Российской Федерации [Электронный ресурс]: URL: `https://cbr.ru/` (дата обращения: 27.02.2025).

2. Единая межведомственная информационная система социального обеспечения (ЕМИСС) [Электронный ресурс]: URL: `https://www.fedstat.ru/` (дата обращения: 27.02.2025).

3. Anselin L. Spatial Econometrics: Methods and Models. The Netherlands: Kluwer Academic Publishers, 1988. 284 p.

4. Lottmann F. Regional unemployment in Germany: a spatial panel data analysis // 52nd Congress of the European Regional Science Association. 2012. P. 1–42.

5. Кадочникова Е. И., Варламова Ю. А. Статистический анализ пространственных данных. Казань: КФУ, 2023. 72 с.

6. Семерикова Е. В., Демидова О. А. Анализ региональной безработицы в России и Германии: пространственно-эконометрический подход // Пространственная экономика. 2015. № 2. С. 64–85.

7. Qi D., Bure V. M. Statistical analysis of investment attractiveness of China's regions // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2022. Т. 18. № 1. С. 188–194.

УДК 519.81

Новгородцев В. А.

О задаче управления портфелем в случае модели Бейтса

Рекомендовано к публикации доцентом Кумачевой С. Ш.

1. Введение. В данной работе описана постановка и предлагается логика решения на основе работ [1, 2] несогласованной во времени задачи [3] стохастического оптимального управления портфелем, содержащей метрику CVaR [4].

В отличие от работ [1, 2], динамика описывается моделью Бейтса [5], т. е. является диффузионным процессом со скачками, а функционал качества, кроме метрики CVaR, содержит функционал из классической постановки.

В работе дана постановка, для которой показано, что исходную задачу можно свести к задаче двухуровневой оптимизации. Далее описывается логика решения с использованием методов [1, 2].

В качестве альтернативных способов решения задач стохастического оптимального управления отметим подходы, связанные с использованием обучения с подкреплением [6, 7].

2. Начальные построения. Пусть $T > 0$. Рассмотрим вероятностное пространство $(\Omega, \mathcal{F}, \mathbb{P})$ с фильтрацией $\{\mathcal{F}_t\}_{t\in[0,T]}$, которая непрерывна справа и такая, что $\mathcal{F}_0$ содержит множества меры нуль.

Рассмотрим совместную динамику n активов в модели Бейтса:

$$\begin{cases} \frac{dS^i(t)}{S^i(t-)} = \mu_i dt + \sqrt{v^i(t)}dW_i^S(t) + (e^{J_i} - 1)dX_{\mathcal{P}_i}(t), \\ dv^i(t) = \kappa_i(\overline{v}_i - v^i(t))dt + \gamma_i\sqrt{v^i(t)}dW_i^v(t), \\ i = 1, \ldots, n, \\ (S^1(0), \ldots, S^n(0))^T = S_0, \\ (v^1(0), \ldots, v^n(0))^T = v_0. \end{cases}$$

где процессы $S = (S^1, \ldots, S^n)$ – стоимости активов; $v = (v^1, \ldots, v^n)$ – волатильности активов; S_0 и v_0 – векторы начальных стоимостей

Новгородцев Виталий Александрович – аспирант, Санкт-Петербургский государственный университет; e-mail: vit.novgorodtcev@mail.ru, тел.: +7(921)771-68-41

и волатильностей активов; μ_i – коэффициент дрифта актива i; $W_1^S, \ldots, W_n^S$ – некоррелированные броуновские движения, адаптированные к фильтрации $\{\mathcal{F}_t\}_{t\in[0,T]}$; $W_1^v, \ldots, W_n^v$ – некоррелированные броуновские движения, адаптированные к фильтрации $\{\mathcal{F}_t\}_{t\in[0,T]}$ и соответствующие волатильностям активов. При этом для каждого актива i имеет место следующая корреляция между броуновскими движениями: $dW_i^S(t)dW_i^v(t) = \rho_{S,v}^i dt$, величина $\kappa_i \geqslant 0$ – скорость возвращения к среднему, $\overline{v}_i \geqslant 0$ – долгосрочное среднее значение процесса волатильности, $\gamma_i > 0$ – волатильность волатильности; $X_{\mathcal{P}_i}$ – пуассоновский процесс с интенсивностью ξ_i и $J_i \sim \mathcal{N}(\mu_{J_i}, \sigma_{J_i}^2)$ – нормально распределенные амплитуды скачков.

Введем множество $\mathcal{A}$ всех допустимых управлений, т. е. $\mathbb{E}\int_0^T \|A_t\|^2\, dt < \infty$, и $\mathcal{F}_t$ – предсказуемых управляемых процессов [8] со значениями в компактном выпуклом множестве $\mathbb{A} = \{a \in \mathbb{R}^n | \sum_{i=1}^n a_i = 1, a_i \geqslant 0\}$. Здесь a_i имеет смысл доли общего капитала в активе i.

Динамика стоимости портфеля X^A [1], составленного из n активов, под воздействием управления $A \in \mathcal{A}$ выглядит следующим образом:

$$\begin{cases} dv^1(t) = \kappa_1(\overline{v}_1 - v^1(t))dt + \gamma_1\sqrt{v^1(t)}dW_1^v(t), \\ \ldots \\ dv^n(t) = \kappa_n(\overline{v}_n - v^n(t))dt + \gamma_n\sqrt{v^n(t)}dW_n^v(t), \\ \frac{dX^A(t)}{X^A(t-)} = A^T(t)\mu dt + A^T(t)\sqrt{v(t)}dW(t) + A^T(t)(e^J - 1)dX_{\mathcal{P}}(t), \\ X^A(0) = x_0, \quad v(0) = v_0, \quad \mu = (\mu_1, \ldots, \mu_n)^T, \\ \sqrt{v(t)}dW(t) = (\sqrt{v^1(t)}dW_1^S(t), \ldots, \sqrt{v^n(t)}dW_n^S(t))^T, \\ (e^J - 1)dX_{\mathcal{P}}(t) = ((e^{J_1} - 1)dX_{\mathcal{P}_1}(t), \ldots, (e^{J_n} - 1)dX_{\mathcal{P}_n}(t))^T. \end{cases} \tag{1}$$

Для любого $A \in \mathcal{A}$ существует единственное решение X^A для уравнения (1) при любых начальных данных [10].

Пусть $\{X_s^{t,x_0,A} | t \leqslant s \leqslant T\}$ – решение уравнения (1) при фиксированных $(t, x_0) \in [0, T] \times \mathbb{R}^+$ и управлении $A \in \mathcal{A}$.

3. Постановка задачи. Рассмотрим задачу минимизации на множестве $\mathcal{A}$ функционала

$$V_{0,x_0} = \inf_{A\in\mathcal{A}}\Big\{\{\mathbb{E}[\int_0^T e^{-rt}h(-X_t^{0,x_0,A}, A(t))dt] + \mathbb{E}[-X_T^{0,x_0,A}]+ \\ + \lambda CVaR_\alpha(-X_T^{0,x_0,A}))\Big\}, \tag{2}$$

$$CVaR_\alpha(\xi) = \inf_{y\in\mathbb{R}} \mathbb{E}[y + \frac{1}{1-\alpha}(\xi - y)^+],$$

где $\alpha \in (0,1)$, $\lambda > 0$, $r > 0$ – дисконтирующий множитель; $A(t) \in \mathbb{A}$ – управление во время t; $h : \mathbb{R}\times\mathbb{A} \to \mathbb{R}$ – липшицева с не более чем квадратичным ростом, совместно выпукла вниз по первому и второму аргументам и неубывающая по первому аргументу функция для удержания процесса в определенном состоянии; индексы $(0, x_0)$ у функции ценности V_{0,x_0} указывают на то, что необходимо знать значение функции ценности для момента времени $t = 0$ и начального состояния процесса $X^A(0) = x_0$.

Используя методологию работы [1], несогласованную во времени задачу (2) сведем к задаче двухуровневой оптимизации вида

$$V_{0,x_0} = \inf_{y\in\mathbb{R}} \overline{V}_{0,x_0}(y),$$

$$\overline{V}_{0,x_0}(y) = \inf_{A\in\mathcal{A}}\Big\{\mathbb{E}[\int_0^T e^{-rt}h(-X_t^{0,x_0,A}, A(t))dt - X_T^{0,x_0,A}+ \\ + \lambda(y + \frac{1}{1-\alpha}(-X_T^{0,x_0,A} - y)^+)]\Big\}. \tag{3}$$

Для ее решения, используя подход из [1], необходимо добиться некоторых дополнительных свойств динамики (1) и функции $f(-X_T^{0,x_0,A}, y) = (y + \frac{1}{1-\alpha}(-X_T^{0,x_0,A} - y)^+)$: равномерной параболичности и равномерной полувогнутости [1]. Для получения этих свойств динамика (1) и функция f возмущаются [1, 9], тогда задача (3) сводится к возмущенной задаче двухуровневой оптимизации с функцией ценности $\overline{V}_{0,x_0,\varepsilon}$, где $\varepsilon > 0$, в которой внутренняя задача является классической, поэтому при существовании достаточно гладкого решения ее можно решать через уравнения Гамильтона – Якоби – Беллмана (ГЯБ) [10]. При этом решение существует при достаточно общих предположениях в вязкостном смысле [10, 11]. В случае, если управление, получаемое через

решение уравнения ГЯБ, будет допустимым, оно будет оптимальным. Тогда задача внешнего уровня может быть решена через градиентные методы, с помощью рассуждений, аналогичных [1, 2].

К дальнейшим перспективам данного исследования относится обобщение результатов, на основании которого возможно построить решение задачи (2) и провести моделирование на практическом примере.

4. Заключение. В данной работе введена постановка несогласованной во времени задачи стохастического оптимального управления портфелем, содержащей метрику CVaR, в которой динамика активов описывается моделью Бейтса, а также описана логика ее решения на основе работ [1, 2]. В дальнейшем планируется реализация этой логики в виде теоремы, а также проведение численного моделирования на реальном примере.

Литература

1. Miller C. W., Yang I. Optimal control of conditional value-at-risk in continuous time // SIAM J. Control Optim. 2017. Vol. 55. No 2. P. 856–884.

2. Kumacheva S., Novgorodtcev V. On the gradient method in one portfolio management problem // Mathematics. 2024. Vol. 12. No 19. P. 3086.

3. Bjork T., Murgoci A. A general theory of markovian time inconsistent stochastic control problems [Электронный ресурс]: URL: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=1694759 (дата обращения: 20.03.2025).

4. Uryasev S., Rockafellar R. T. Conditional value-at-risk: optimization approach // Stochastic Optimization: Algorithms and Applications. Applied Optimization. 2001. Vol. 54. P. 411–435.

5. Bates D. S. Jumps and stochastic volatility: exchange rate processes implicit in deutsche mark options // The Review of Financial Studies. 1996. Vol. 9. No 1. P. 69–107.

6. Новгородцев В. А., Кудин П. С., Кумачева С. Ш. Об оптимизации решений с помощью прогнозного модуля в обучении с подкреплением // Процессы управления и устойчивость. 2023. Т. 10. № 1. С. 454–461.

7. Wang H., Zhou X. Y. Continuous-time mean–variance portfolio selection: A reinforcement learning framework // Mathematical Finance. 2020. Vol. 30. No 4. P. 1273–1308.

8. Ширяев А. Н. Основы стохастической финансовой математики: В 2–х т. Т. 1: Факты, модели. М.: МЦНМО. 2016. 440 с.

9. Полякова Л. Н. Smooth approximations of nonsmooth convex functions // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2022. Т. 18. № 4. С. 535–547.

10. Shi J. T., Wu Z. Relationship between MP and DPP for the stochastic optimal control problem of jump diffusions // Appl. Math. Optim. 2011. Vol. 63. P. 151–189.

11. Nikitin F. F. Viscosity solutions and programmed iteration method for Isaacs equation // Вестник Санкт-Петербургского университета. Серия 10: Прикладная математика. Информатика. Процессы управления. 2014. № 2. P. 84–92.

УДК 519.86

Субботина Е. А.

Оптимизация затрат на рекламу в социальных сетях с использованием моделей влияния

Рекомендовано к публикации доцентом Лежниной Е. А.

1. Введение. Пользователи или агенты взаимодействуют в социальных сетях и влияют друг на друга, а иногда даже побуждают друг друга к действиям. Анализ этих влияний, а также выявление самых влиятельных агентов может помочь в проведении наиболее эффективных рекламных кампаний. В представленной работе на примере данных социальной сети «ВКонтакте» будут выявлены наиболее влиятельные из группы агентов и определена доля тех, кому следует выдать бесплатные образцы продукции, чтобы добиться желаемого результата рекламной кампании.

2. Постановка задачи. Пусть компания, организующая юмористические концерты, хочет продвигать свои услуги в социальной сети с ограниченным бюджетом. Планируется раздать бесплатные билеты на концерт блогерам, чтобы они поделились впечатлениями и повлияли на целевую аудиторию. Необходимо выбрать группу пользователей для раздачи бесплатных образцов, чтобы треть аудитории в результате приобрела билеты.

3. Математическая модель. Будем использовать математическую модель, описанную в [1].

3.1. Матрица влияния. Пусть в нашей социальной сети n агентов. Степени их влияния друг на друга определим $(n \times n)$-матрицей A с элементами $a_{ij} > 0$, которые обозначают силу влияния i-го агента на j-го.

Величина a_{ij} зависит от многих факторов, например, такими факторами могут быть общие интересы, дружеские отношения и т. д. Обозначим их как $x_{ij}^l \in [0, 1], l = 1, \ldots, 5$ (условие достигается с помощью нормировки). В рамках данной работы в качестве факторов

Субботина Екатерина Андреевна – магистрант, Санкт-Петербургский государственный университет; e-mail: st087998@student.spbu.ru, тел.: +7(911)091-81-89

влияния выберем общих подписчиков, общие подписки, общих друзей, отношения агентов (1 – если i-й агент есть в друзьях у j-го, в противном случае – 0) и количество репостов у каждого из агентов со страницы другого.

Далее по формуле

$$a_{ij} = \sum_{l=1}^{5} \alpha_l x_{ij}^l, \tag{1}$$

где $\alpha_{ij}^l \in [0,1], l = 1, \ldots, 5,$ – весовой вектор, найдем элементы матрицы влияния A. Построенная матрица A – матрица смежности некоторого ориентированного графа $G(A)$. Если влияние i-го агента на j-го слишком мало, будем считать, что пути из i в j нет.

3.2. Иерархическая кластеризация агентов влияния. Построим матрицу T транзитивного замыкания полученного графа $G(A)$ по алгоритму Флойда–Уоршелла. Для этого воспользуемся формулой

$$t_{ij} = t_{ij} \vee (t_{ik} \wedge t_{kj}) \quad \forall i, j, k = \overline{1, n}. \tag{2}$$

Затем воспользуемся алгоритмом нахождения канонической формы полученной матрицы T [2]. Если T – матрица, состоящая из единиц, то граф $G(A)$ – сильно связный, а матрица A – неприводима [3]. В остальных случаях составляем вектор S, состоящий из сумм строк матрицы T. Далее строится матрица перестановок P такая, что $PS = \widetilde{S}$, где $\widetilde{S}$ – вектор, содержащий значения вектора S, отсортированные в порядке убывания.

Используя полученные результаты, найдем итоговую матрицу влияния

$$\widetilde{A} = PAP^T. \tag{3}$$

В результате получим m кластеров $C_1, \ldots, C_m$ с $n_i, i = \overline{1, k}$, элементами в каждом.

3.3. Определение необходимой доли агентов. Воспользуемся пороговой моделью q-влияния, описанной в [4].

Пусть компания предоставляет бесплатные образцы $\rho < 1$ доле агентов социальной сети. Пользователи, получившие бесплатный образец, купят продукт по собственному желанию с вероятностью p^+, а не получившие – также могут купить продукт с вероятностью p^-, где $p^+ > p^-$. Пользователи, купившие продукт, могут повлиять на своих знакомых, чтобы они тоже купили его с вероятностью q.

Пусть φ_i – случайная величина, имеющая распределение Бернулли. Она принимает значение 1, если агент i решил купить продукт по собственному желанию, т. е. $P(\varphi_i = 1) = p^+$, и значение 0, если пользователь приобрел товар под влиянием рекламы, т. е. $P(\varphi_i = 0) = p^-$. Тогда φ_i имеет параметр

$$\mu = \rho p^+ + (1 - \rho)p^-. \tag{4}$$

Таким образом

$$1 - E[Y] = (1 - \mu) \sum_{k=0}^{\infty} P_1(k + 1)(1 - qE[Y])^k, \tag{5}$$

$$1 - E[X] = (1 - \mu) \sum_{k=0}^{\infty} P_0(k)(1 - qE[Y])^k. \tag{6}$$

Здесь X – случайная величина, показывающая, купил ли продукт некоторый узел r, Y – случайная величина, которая демонстрирует, приобретет ли товар некорневой узел i. Данные величины распределены в соответствии с распределением Бернулли.

Величина $P_1(k)$ – это вероятность того, что дочерний узел имеет степень k, а $P_0(k)$ – вероятность того, что корневой узел имеет степень k. Эти значения вычисляются по формулам:

$$P_0(k) = \frac{k^{-\gamma}}{\zeta(\gamma)}, \quad P_1(k) = \frac{k^{1-\gamma}}{\zeta(\gamma - 1)}, \quad k = 1, 2, \ldots$$

Здесь константа $2 < \gamma < 3$, $\zeta(x)$ – дзета-функция Римана.

4. Пример. Решим задачу на примере реальных данных социальной сети «ВКонтакте». Выберем 20 блогеров, состоящих в популярном сообществе «Импроком», специализирующемся на создании юмористического контента. Используя API «ВКонтакте», получим данные для заполнения матриц факторов влияния. После нормировки и задания весового вектора $\alpha = [0{,}75; 0{,}8; 0{,}65; 0{,}5; 0{,}5]$ по формуле (1) найдем все элементы матрицы влияния A. Визуализируем полученный результат, построив граф $G(A)$ (см. рис. 1).

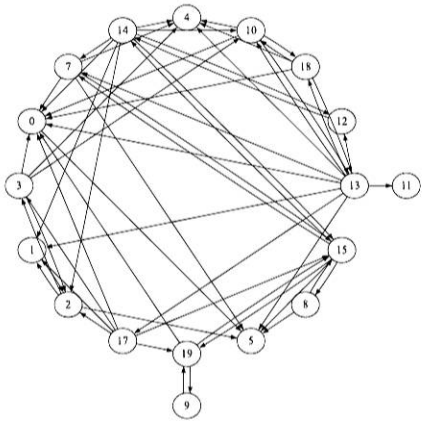


Рис. 1. Граф $G(A)$

Следуя алгоритму, описанному в пункте 3.2, и используя формулы (2), (3), получим иерархическую кластеризацию агентов влияния (см. рис. 2).

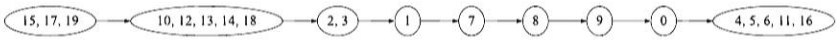


Рис. 2. Иерархическая кластеризация агентов влияния

По полученным результатам, можно найти среднее значение влияния q на каждого из агентов сети:

$$\overline{q} = \frac{1}{n}\sum_{j=1}^{n}\sum_{i=1}^{n} a_{ij} = \frac{1}{20}8{,}24 \approx 0{,}41.$$

Теперь согласно формулам (5), (6) можно подсчитать параметр $E[X]$ для различных μ. Требуемой по условию задачи доле пользователей $E[X] \approx 0{,}33$ соответствует $\mu \approx 0{,}1$. Тогда из формулы (4) выразим искомое значение

$$\rho = \frac{\mu - p^-}{p^+ - p^-} = \frac{0{,}1 - 0{,}05}{0{,}3 - 0{,}05} = 0{,}2.$$

Здесь $p^+ = 0{,}3$ и $p^- = 0{,}05$ – вероятности приобретения билета пользователями, получившими и не получившими бесплатный образец соответственно. Тогда необходимо раздать бесплатные билеты $\rho \times n = 0{,}2 \times 20 = 4$ наиболее влиятельным пользователям. Согласно

рис. 2 таковыми являются пользователи под номерами 15, 17 и 19. Для выбора четвертого агента обратимся к кластеру C_2 и повторно применим к нему алгоритм иерархической кластеризации. Результат представлен на рис. 3.

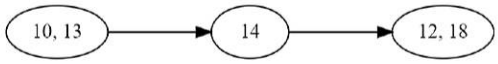


Рис. 3. Иерархическая кластеризация агентов влияния внутри кластера C_2

Из агентов 10, 13 выберем, например, пользователя под номером 13. Таким образом для раздачи бесплатных билетов выберем агентов под номерами 13, 15, 17, 19.

5. Заключение. В работе решена задача определения набора пользователей для раздачи бесплатных образцов продукции с целью привлечения трети агентов социальной сети к покупке товара. На языке Python реализована программа, моделирующая взаимоотношения агентов на основе реальных данных. Предложенный подход позволяет оптимизировать маркетинговые стратегии, сократить расходы и повысить прибыльность бизнеса. Результаты, основанные на реальных данных, применимы в конкретных бизнес-сценариях для достижения конкурентных преимуществ.

Литература

1. Лежнина Е. А., Хитров Г. М., Калинина Е. А., Кононов И. А. Рабочие группы: создание и управление // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2022. Т. 18. №. 4. С. 608–620.
2. Савицкая Д. В. Нормальная форма (0,1)-матрицы и алгоритмы ее построения // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2008. № 4. С. 85–98.
3. Guo H., Li M., Shuai Zh. Global stability of the endemic equilibrium of multigroup SIR epidemic models // Canadian Applied Mathematics Quarterly. 2006. P. 259–284.
4. Zhao B., Li Y. K., Lui J. C. S., Chiu D.-M. Mathematical modeling of advertisement and influence spread in social networks // In NetEcon: Workshop on the Economics of Networks, Systems and Computation. 2009.

УДК 510

Фролова Е. А.

Модель энергетического микса в Российской Федерации

Рекомендовано к публикации доцентом Балыкиной Ю. Е.

1. Введение. Энергетическая система играет ключевую роль в экономике и жизнеобеспечении. Россия – один из крупнейших производителей, потребителей и экспортеров углеродных ресурсов, а также лидер в атомной энергетике и гидроэнергетике. В условиях глобального перехода к низкоуглеродной энергетике страны стремятся к сокращению выбросов CO_2. Мировой энергетический ландшафт меняется, а источники возобновляемой энергии становятся все более востребованными [1–3]. Россия участвует в международных климатических инициативах, ведет собственную политику в области энергетики и разрабатывает национальные программы по сокращению выбросов CO_2.

2. Актуальность. Согласно распоряжению Правительства РФ от 9 июня 2020 года № 1523-р об утверждении энергетической стратегии на период до 2035 года будут сохраняться текущие темпы роста спроса на электроэнергию, доли выработки электроэнергии на основе атомной энергии, гидроэнергии и возобновляемых источников будут увеличиваться, а доли выработки на основе нефти, газа и угля сокращаться. Также в стратегии учитывается экологический фактор, и установлены предельные значения по выбросам CO_2. Для достижения этих целей необходимо оптимизировать структуру энергопотребления, сократить углеродный след и эффективно распределить ресурсы.

3. Постановка задачи и методы ее решения. В данной работе проведен анализ текущей структуры энергетики России. На основе исторических данных о потреблении электроэнергии с 1995 по 2023 год с помощью метода линейной регрессии был построен прогноз общего спроса в стране на 2028 год. Была разработана модель,

Фролова Елизавета Алексеевна – магистрант, Санкт-Петербургский государственный университет; e-mail: st075663@student.spbu.ru, тел.: +7(921)940-69-12

позволяющая строить сценарные прогнозы распределения энергии в зависимости от целевых показателей, между источниками: нефть, газ, уголь, атомная энергия, гидроэнергия, солнечная энергия и ветровая.

4. Модель энергетического баланса на 2028 год. Все данные были взяты из открытых источников, предоставляющих статистические показатели, а также официальные отчеты и прогнозы в области энергетики [4].

4.1. Целевая функция. Модель формализуется, как задача линейного программирования с одной целевой функцией и набором ограничений [5]. Целью оптимизации является минимизация совокупных затрат на производство электроэнергии.

$$\min Z = \sum_{i \in X} c_i x_i, \tag{1}$$

где X – множество источников энергии (нефть, газ, уголь, атомная энергия, гидроэнергия, солнечная энергия и ветровая); c_i – стоимость производства 1 ТВт·ч электроэнергии источником i; x_i – объем электроэнергии, произведенной источником i (ТВт·ч) за год.

4.2. Ограничения. В модели вводятся следующие ограничения:

1. Баланс спроса и предложения

$$\sum_{i \in X} x_i = D, \tag{2}$$

где D – прогнозируемый общий спрос на электроэнергию в 2028 году (ТВт·ч).

2. Ограничения на установленную мощность

$$x_i \leqslant P_i F_i, \quad \forall i \in X, \tag{3}$$

где P_i – установленная мощность источника (ГВт), F_i – коэффициент использования установленной мощности для источника i.

Коэффициент использования определяет долю от установленной мощности для каждого источника: нефть (50%), газ (80%),

уголь (70%), атомная энергия (90%), гидроэнергия (60%), солнечная энергия (20%), ветровая энергия (40%).

3. Ограничение по выбросам CO_2

$$\sum_{i \in X} E_i x_i \leqslant E_{\max}, \tag{4}$$

где E_i – удельные выбросы CO_2 (млн тонн/год) для источника i; $E_{\max}$ – предельный уровень выбросов CO_2, допустимый в 2028 году.

4. Минимальная доля каждого источника в общем производстве (для обеспечения энергетической диверсификации).

Для того, чтобы модель использовала все виды источников в оптимальном энергобалансе, вводится параметр α_i, который определяет минимальную долю производства электроэнергии каждым источником. В модели α_i составляет 5% для всех источников.

$$x_i \geqslant \alpha_i D, \quad \forall i \in S. \tag{5}$$

5. Результаты. В рамках исследования было рассмотрено 3 сценария: в первом сохраняются текущие темпы роста спроса, во втором рассматривалось падение спроса, в третьем увеличение спроса. Проведенное моделирование позволило получить энергетическую структуру Российской Федерации на 2028 год. В таблице в качестве примера представлены результаты, полученные для сценария с сохранением текущих темпов роста спроса. Общие затраты на 2028 год были рассчитаны в соответствии с ценами на производство 1 ТВт·ч электроэнергии в 2023 году. Таким образом, в 2023 году общие затраты на производство электроэнергии составили 72 млрд долларов, а к 2028 году затраты увеличатся до 82 млрд долларов.

Диаграмма на рис. 1 иллюстрирует энергетическую структуру 2023 года. В 2023 году общий объем потребления электроэнергии составил 1111 ТВт·ч (см. таблицу), при этом основными источниками были газ (46%), уголь (15%) и атомная энергия (19%). Доля возобновляемых источников была незначительна (солнечная энергия – 0,2% и ветряная – 0,6%).

Таблица. Распределение энергии между источниками в 2023 году и результаты на 2028 год

Производство ТВт·ч	2023 год	2028 год
Нефть	8,5	31,0
Газ	513,0	211,0
Уголь	165,0	157,0
Атомная	217,0	356,4
Гидроэнергия	198,0	292,0
Солнечная	2,7	113,0
Ветряная	7,6	92,0
Общий спрос ТВт·ч/год	1111,8	1252,0
Общие затраты млрд $	72,4	82,9



Рис. 1. Доля производства электроэнергии по источникам 2023 год

Диаграмма на рис. 2 представляет энергетическую структуру 2028 года. Видно, что для достижения целевых показателей, должна существенно увеличиться доля возобновляемых источников: доля солнечной энергии увеличится до 9%, а ветряная – до 7%. Доля газа в общем производстве снизится до 17%, а угля – до 12%. Атомная энергия увеличит свою значимость с долей в 29%, а доли нефти, газа и угля снизятся, что соответствует утвержденной стратегии в РФ.

6. Выводы. Результаты проведенного моделирования показали, что к 2028 году ожидается увеличение доли возобновляемых источников энергии и доли атомной энергии в структуре энергобаланса РФ, при этом доля ТЭС сократится. Ожидается значительное увеличение доли возобновляемых источников энергии, таких как солнечная и ветровая, при этом доля атомной энергии увеличится, а доля

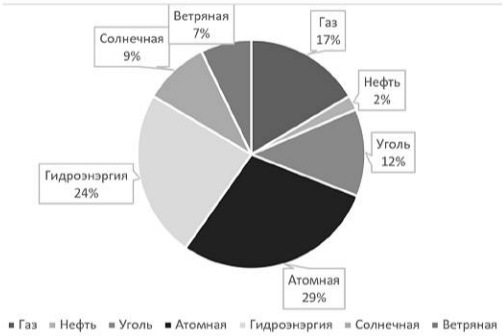


Рис. 2. Доля производства электроэнергии по источникам 2028 год

ТЭС сократится, что соответствует макроэкономическим трендам, политике декарбонизации и прогнозам технологического развития.

Литература

1. Negishi S. A study of long-term energy-mix optimization model: a case study in Japan [Электронный ресурс]: URL:https://www.researchgate.net/publication/362088797 (дата обращения: 11.04.25).
2. Weijermars R., Taylor P. Review of models and actors in energy mix optimization – can leader visions and decisions align with optimum model strategies for our future energy systems? // Energy Strategy Reviews. 2012. P. 5–18.
3. Яхья М. А. К., Головкина А. Г. Математическое моделирование и оптимальное управление энергопотреблением микросети комбинированной выработки теплаи электроэнергии // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2024. Т. 20. № 4. С. 446–466.
4. Statistical review of world energy [Электронный ресурс]: URL: https://www.energyinst.org/statistical-review (дата обращения: 20.02.25).
5. Gruenwald O. Optimization model of energy mix taking into account the environmental impact // 11th International Scientific Conference on Control of Power Systems. Bratislava: Slovak University of Technology in Bratislava, 2012. P. 1–8.

UDC 519.83

Cao Z.

Dynamic networks with complete information and incomplete information under cross-partitioning

Recommended for publication by professor Li Y.

1. Introduction. The theoretical development of dynamic network formation models has been profoundly influenced by foundational works such as pairwise stability by Jackson and Wolinsky [1] and Nash equilibrium networks by Bala and Goyal [2]. Recent advances include Sun and Parilina's ordered partitioning model with asymmetric players [3], which demonstrates the nonlinear impact of inter-group label distances on connection costs. Their simulations further reveal that incomplete information induces significant topological heterogeneity in stable networks. Their subsequent work [4] shows that the decay factor δ critically shapes network convergence dynamics. However, existing models predominantly assume mutually exclusive group memberships, which contradicts real-world scenarios where agents often belong to overlapping communities (e.g., cross-organizational collaboration).

Building on framework from [4], we propose a dynamic network formation model with overlapping groups. By refering Galeotti et al.'s incomplete information game theory [5], under their contributions, we define a utility function based on minimal inter-group distances. And do the simulation of overlapping groups.

2. Overlapping groups, functions and dynamic network formation process. This section describes the functions and dynamic network formation processes required for cross-grouping and experimentation.

2.1. Network and player cross-partitioning. Consider a finite set of players $N = \{1, \ldots, n\}$. The communication structure is represented by a network $g \subseteq G = \{ij \mid i, j \in N,\ i \neq j\}$. Players belong to overlapping groups $N_1, \ldots, N_m$ (for example, players are divided into $[[1, 2, 3], [3, 4, 5]]$. So the player 3 is the overlapping part of two groups). The distance $|\pi(i) - \pi(j)|$ captures inter-group distance, influencing link costs.

Cao Zeyu – graduate student, Yan'an University; e-mail: bba004822@gmail.com

2.2. Player utility. Player i's utility in network g is:

$$\Pi_i(g) = \sum_{i \overset{g}{\leftrightarrow} j} \delta^{d_{ij}(g)-1} - \sum_{ij \in g} f\left(\min\{|l-k|\}\right)$$

where $\delta \in (0,1]$ is a decay factor, $l \in \pi(i)$, $k \in \pi(j)$, and f is a non-decreasing cost function.

2.3. Expected unit cost. Player i's expected unit cost in network g is:

$$E_i = \sum_{k=1}^{m} h_i(k) f\left(\min\{|k-l|\}\right)$$

$h_i(k)$ is a prior belief that is a guess probability of the opposing player's group when we don't know the information about the opponent's group.

2.4. Dynamic network formation process. Starting from an empty network, a randomly selected pair (i,j) updates their link each stage. Players act myopically to maximize expected utility. Under incomplete information, beliefs $h_i(k)$ about others' group memberships guide decisions. Beliefs update upon connection using a simple rule.

3. Stable equilibria and classification of information.

3.1. Stable equilibrium. A strategy profile $s^* = (s_1^*, \ldots, s_n^*)$ forms a stable equilibrium if

$$\mathbb{E}\left[\Pi_i\Big(\left(W_i(\sigma) \cup W_j(\sigma)\right) + ij\Big) \mid B_i(\sigma)\right] \geq \Pi_i\left(W_i(\sigma) \setminus ij\right),$$

where the left-hand side equals $\Pi_i\left(\left(W_i(\sigma) \cup W_j(\sigma)\right) + ij\right)$ if $j \in B_i(\sigma)$, or $\Pi_i\left(\left(W_i(\sigma) \cup W_j(\sigma)\right) + ij\right) + f\Big(\min\{|l-k|\}\Big) - E_i$ if $j \notin B_i(\sigma)$ and $j \in N_K$, for all i, j, and histories which is about connections σ. Initially, players only know the tags of the other players in their group or groups (if coming from more than one group).

3.2. Complete vs. incomplete information. Under complete information, players know all group labels initially. Under incomplete information, Initially, the player only knows the tags of other players in his group (if he comes from more than one group, he knows all of his group tags), beliefs updated through interactions.

4. Simulation experiments. In this part, we do experiments on the stabilization time under the cross-group division.

4.1. Stabilization time. Experiments compare stabilization times for $n = 7$ players with overlapping groups $[1, 2, 3]$, $[3, 4, 5]$, $[4, 5, 6]$, $[5, 6, 7]$. To simplify the experiment, a set of expected unit costs and true costs were fixed as $f(0) = \frac{1}{2}$, $f(1) = \frac{8}{5}$, $f(2) = 3$, $f(3) = 4$, $E_i = 0.95$, $i = 1, \ldots, 7$. For $\delta = 1$ the stabilization time is shown at Figure 1 and the stabilized networks are shown at Figure 2 (numbers in the circles mean the players). The corresponding results for $\delta = 0.7$ are presented at Figures 3 and 4.

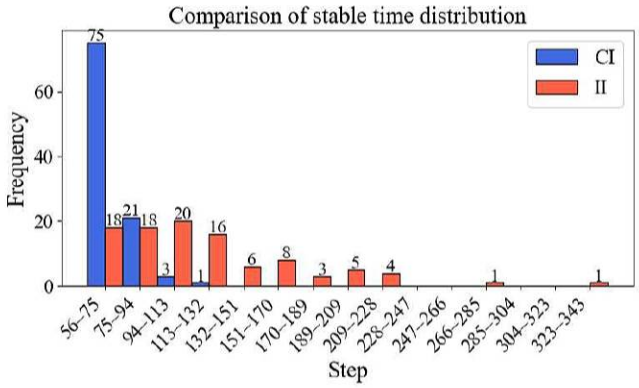


Figure 1. Stabilization time distributions under complete (blue) and incomplete (red) information for $\delta = 1$

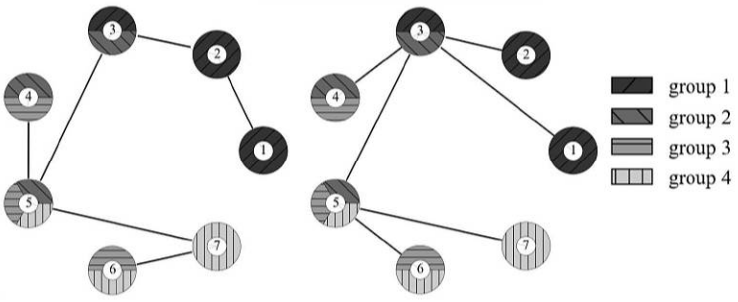


Figure 2. Stabilized networks under complete (left) and incomplete (right) information for $\delta = 1$

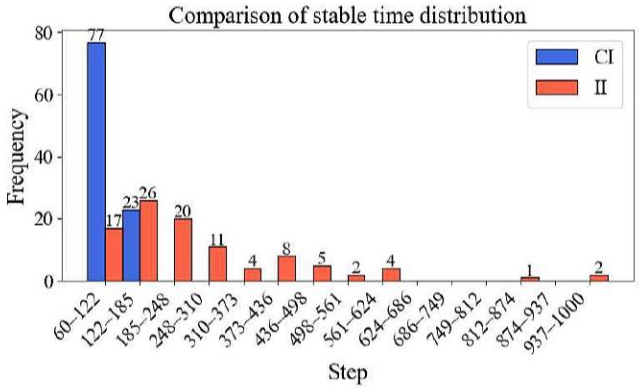


Figure 3. Stabilization time under complete (blue) and incomplete (red) information for $\delta = 0.7$

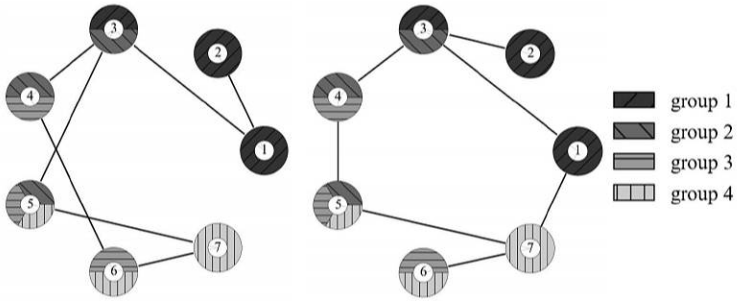


Figure 4. Stabilized networks under complete (left) and incomplete (right) information ($\delta = 0.7$)

5. Conclusion. According to the experimental results, distribution diagrams at Fig. 1 (for $\delta = 1$) and 3 (δ=0.7) show that the stable time distribution interval of the network under complete information is smaller than the stable time distribution interval under incomplete information. We also can see that the greater is the revenue decay (δ), the longer is the stabilization time.

References

1. Jackson M. O., Wolinsky A. A strategic model of social and economic networks // Journal of Economic Theory. 1996. Vol. 71. No 1. P. 44–74.

2. Bala V., Goyal S. A noncooperative model of network formation // Econometrica. 2000. Vol. 68. No 5. P. 1181–1229.

3. Sun P., Parilina E. Impact of utilities on the structures of stable networks with ordered group partitioning // Dynamic Games and Applications. 2022. Vol. 12. P. 1131–1162.

4. Sun P., Parilina E. Dynamic network formation with ordered partitioning and incomplete information // Dynamic Games and Applications. 2024. Vol. 14. P. 921–958.

5. Galeotti A., Goyal S., Jackson M. O., Vega-Redondo F., Yariv L. Network games // Review of Economic Studies. 2010. Vol. 77. No 1. P. 218–244.

UDC 519.83

Du L.

The stability of competing information flow in epidemic models

Recommended for publication by associate professor Gubar E. A.

1. Introduction. The dissemination of information during critical events, such as pandemics or political campaigns, significantly influences public opinion and behavior. Understanding how different types of information are spread is crucial for managing their impact.

The Susceptible–Infected–Recovered (SIR) model [1], originally used to describe disease spread, is adapted here to account for the strategic choices individuals make about which information to believe and share. Building on recent work [2], we unify information dissemination and strategic decision-making into a single dynamical system. By incorporating evolutionary game theory and replicator dynamics, we analyze the system's stability and identify conditions under which one type of information dominates or coexists with the other. Our primary contribution is a stability analysis using Lyapunov functions and equilibrium analysis, which reveals conditions for stable equilibria. These findings provide insights into managing the spread of competing information types.

2. Model setup and formulation. In epidemiological dynamics, the SIR model is a classical model of transmission of diseases. Divides the total population into three categories: Susceptible (S), Infectious (I), and Recovered (R). The transmission process is described by the following differential equations:

$$\frac{dS}{dt} = -\beta SI, \qquad \frac{dI}{dt} = \beta SI - \sigma I, \qquad \frac{dR}{dt} = \sigma I. \tag{1}$$

Here, β is the infection rate, representing the probability of a susceptible individual becoming infectious after contact, and σ is the recovery rate, indicating the rate at which infectious individuals recover.

In the SIR model, disease transmission is influenced not only by biological factors but also by information dissemination. According to [2],

Du Lei – graduate student, Saint Petersburg State University; e-mail: st122210@student.spbu.ru, phone: +7(812)363-65-37

individuals in a population are exposed to two competing types of information, represented by x_1 and x_2. Each individual acts as a player in a strategic setting, where their strategy is the choice of which type of information to follow. This decision influences their behavior, such as whether to adopt protective measures or engage in risky interactions, ultimately affecting the disease transmission dynamics.

The payoff matrix reflects the costs associated with following each type of information, specifically given by:

$$A(I) = I \begin{pmatrix} c_1 & k_1 \\ k_2 & c_2 \end{pmatrix} + (1 - I) \begin{pmatrix} \bar{c}_1 & \bar{k}_1 \\ \bar{k}_2 & \bar{c}_2 \end{pmatrix},$$

where c_i and $\bar{c}_i$ denote the treatment costs for following different types of information in the infected and non-infected states, k_i and $\bar{k}_i$ denote the costs experienced by people following different types of information in the two states during the information dissemination process.

The time evolution of the proportions of the two types of information readers is described by the following equations:

$$\dot{x_1} = (p(e_1, X) - p(X, X))\, x_1, \quad \dot{x_2} = (p(e_2, X) - p(X, X))\, x_2, \qquad (2)$$

where $X = (x_1, x_2)$, $e_1 = (1, 0)$, $e_2 = (0, 1)$ denote pure strategies and $p(\cdot, \cdot)$ is the payoff function. The expected payoff of the i-th strategy of the first player is given by $p\,(e_1, X) = e_1 A\,(I)\, X$, $p\,(e_2, X) = e_2 A\,(I)\, X$. Ultimately, (2) takes the following specific form:

$$\begin{aligned} \dot{x}_1 = {} & x_1^2 x_2 \left(\bar{c}_1 - \bar{k}_2 + I\left(c_1 - \bar{c}_1 - k_2 + \bar{k}_2\right)\right) + \\ & + x_1 x_2^2 \left(\bar{k}_1 - \bar{c}_2 + I\left(k_1 - \bar{k}_1 - c_2 + \bar{c}_2\right)\right). \end{aligned} \qquad (3)$$

Due to the complexity of the information system model, we denote: $C = \bar{c}_1 - \bar{k}_2 + I(c_1 - \bar{c}_1 - k_2 + \bar{k}_2)$, $\quad D = \bar{k}_1 - \bar{c}_2 + I(k_1 - \bar{k}_1 - c_2 + \bar{c}_2)$. Here, C and D are merely symbols and have no actual physical meaning.

3. Analysis of the model. The system (3) admits three equilibrium solutions, representing the long-term behavior of the population under different conditions, with a fixed infection rate I value.

1. Boundary Equilibrium: $(x_1 = 1, x_2 = 0)$ and $(x_1 = 0, x_2 = 1)$. These represent scenarios where the population is entirely biased towards

one type of information.

2. Interior Equilibrium: $x_1^* = \frac{D}{D-C}$. This equilibrium exists if and only if $C \neq D$, $CD \neq 0$, and $x_1^* \in (0,1)$. It represents a state where both types of information coexist in the population.

Next, we will analyze the stability of the three equilibrium solutions derived earlier. We used the Lyapunov function method [3] to assess the stability of the boundary equilibria. The candidate Lyapunov function is defined as:

$$V(x_1) = \frac{1}{2}(x_1 - x_1^e)^2, \tag{4}$$

where x_1^e is the equilibrium point being studied. For the boundary equilibria: for $(x_1 = 1, x_2 = 0)$, $x_1^e = 1$; for $(x_1 = 0, x_2 = 1)$, $x_1^e = 0$.

The time derivative of the Lyapunov function is:

$$\dot{V}(x_1) = (x_1 - x_1^e)\, x_1 (1 - x_1) [x_1 C + (1 - x_1) D]. \tag{5}$$

Theorem 1. *The stability conditions are:*

1. *if $C > 0$, then the equilibrium $(1,0)$ is asymptotically stable;*
2. *if $D < 0$, then the equilibrium $(0,1)$ is asymptotically stable.*

To analyze the stability of the internal equilibrium, we use the Lyapunov function:

$$V(x_1, x_2) = \sum_{i \in \{1,2\}} x_i^* \log \frac{x_i^*}{x_i}, \tag{6}$$

where x_1^* and x_2^* represent the equilibrium values. The time derivative of the Lyapunov function is given by

$$\dot{V}(x_1) = \frac{[x_1 C + (1 - x_1) D]^2}{C - D}. \tag{7}$$

To make the interior equilibrium asymptotically stable, it is required that $\dot{V}(x_1) < 0$, which implies $C - D < 0$. Combining this with the existence condition of interior equilibrium $0 < x_1^* < 1$, then:

Theorem 2. *If $C < 0$ and $D > 0$, then the interior solution is asymptotically stable.*

Since the parameters C and D depend on k, which in turn can be defined by c, the stability conditions can be further expressed in terms of c. The next step is to analyze the effect of c on the parameter k.

Case 1: $c_1 > c_2$. The parameter k falls within the range:

$$k_1 \in \left[c_2 + \frac{1-I}{I}(\bar{c}_2 - \bar{k}_1), c_1\right], \; k_2 \in \left[\max\{c_2, c_1 + \frac{1-I}{I}(\bar{c}_1 - \bar{k}_2)\}, c_1\right],$$

$$\bar{k}_1 \in \left[\max\{\bar{c}_1, \bar{c}_2 - \frac{I}{1-I}(c_1 - c_2)\}, \bar{c}_2\right], \quad \bar{k}_2 \in [\bar{c}_1, \bar{c}_2].$$

Case 2: $c_1 < c_2$. The parameter k falls within the range:

$$k_1 \in \left[\max\{c_1, c_2 + \frac{1-I}{I}(\bar{c}_2 - \bar{k}_1)\}, c_2\right], \; k_2 \in \left[c_1 + \frac{1-I}{I}(\bar{c}_1 - \bar{k}_2), c_2\right],$$

$$\bar{k}_1 \in [\bar{c}_2, \bar{c}_1], \quad \bar{k}_2 \in \left[\max\{\bar{c}_2, \bar{c}_1 - \frac{I}{1-I}(c_2 - c_1)\}, \bar{c}_1\right].$$

Thus, given a set of parameters c and I, the range of k is determined. Selecting any k within this range ensures the existence of an internal equilibrium that is asymptotically stable.

4. Numerical solution of the internal equilibrium. To validate the theoretical results from the previous section, we numerically solve the equations for different parameter conditions using Python. Game-theoretic simulations have been widely used to model strategic interactions in epidemic spreading, as reviewed in [4]. We apply similar methods to verify our findings and analyze the system's stability. The parameters used are $c_1 = 0.56$, $c_2 = 0.6$, $k_1 = 0.57$, $k_2 = 0.57$, $\bar{c}_1 = 0.08$, $\bar{c}_2 = 0.06$, $\bar{k}_1 = 0.08$, $\bar{k}_2 = 0.08$, $I = 0.3$.

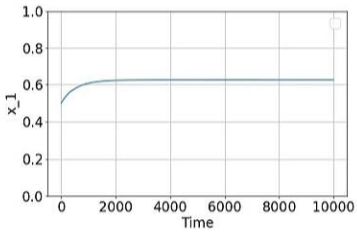


Figure 1. A stable solution

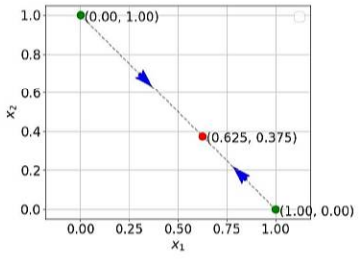


Figure 2. Phase portrait

Figure 1 shows the evolution of x_1 over time t for a specific set of

parameters, based on the conclusions of the previous section. As seen in the figure, the equilibrium solution under these parameters is stable.

A phase portrait is an important tool for analyzing dynamical systems, illustrating the evolution of solutions in the state space [5]. It helps in studying system stability, periodic behavior, and equilibrium points. Figure 2, based on the parameters in Figure 1, shows the system's trajectories under different initial conditions. In this phase portrait, the boundary equilibria $(1, 0)$ and $(0, 1)$ are located at the edges of the coordinate axes, while the internal equilibrium $(0.625, 0.375)$ is also marked. The trajectories indicate that as time progresses, the system evolves toward the internal equilibrium. The direction field arrows point inwards, confirming the stability of this equilibrium. Meanwhile, the boundary equilibria are unstable, consistent with the analysis presented in previous sections.

References

1. Kermack W. O., McKendrick A. G. A contribution to the mathematical theory of epidemics // Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character. 1927. Vol. 115. No 772. P. 700–721.

2. Gubar E., Taynitskiy V., Dahmouni I. The impact of fake news on infection dynamics in Pandemic control: An evolutionary SIR model // IFAC-PapersOnLine. 2023. Vol. 56. No 2. P. 1778–1783.

3. Lyapunov A. M. The general problem of the stability of motion // International Journal of Control. 1992. Vol. 55. No 3. P. 531–534.

4. Huang Y., Zhu Q. Game-theoretic frameworks for epidemic spreading and human decision-making: A review // Dynamic Games and Applications. 2022. Vol. 12. P. 7–48.

5. Poincaré H. Mémoire sur les courbes définies par une équation différentielle // Journal de Mathématiques Pures et Appliquées. 1881. Vol. 7. P. 375–422.

UDC 519.83

Kong Q., Petrosyan L. A.

Dynamic network game with changing communication structure

1. Introduction. Modern network interactions in social and economic systems increasingly exhibit time-dependent connectivity patterns, challenging traditional static game-theoretic assumptions. While foundational work by Petrosyan et al. [1] established instability for cooperative communication structures, the crucial dimension of network evolution remains underexplored in cooperative game theory. This paper bridges this critical gap by developing a dynamic network game framework with co-evolving strategies and communication structures. The principal contribution of this study lies in its pioneering demonstration of dynamic network restructuring in cooperative gaming environments – a critical advancement beyond conventional static analyses.

2. Dynamic network game. Consider an n-person dynamic network game over the time horizon $[t_0, T]$. Each player is represented as a node in the network, and $N = \{1, 2, \ldots, n\}$ denotes the set of players. For $i, j \in N$, a connection between i and j is represented by an arc $(i, j) \in L$, where L is the set of all arcs in the network.

Then the set of connections of the player i can be denoted as

$$K(i) = \{j : \text{arc}(i, j) \in L\}, \quad K(i) \in N, \quad i \notin K(i).$$

At t_0, players simultaneously and independently choose neighbors with whom they intend to interact during the game. Player i can choose neighbors from a fixed subset of players $N_i \subset N \setminus \{i\}$, where N is the set of all players. The sets N_i can be different for different players, and for each player i, the number of its possible neighbors is limited by the number k_i. Communication is established (that is, the link is created in the network) between players i and j if $i \in N_j$ and $j \in N_i$.

Due to connection constraints, the same set of players can form different networks. The set of networks is denoted as $\mathbf{G} = \{G_m \mid m =$

Kong Qi – graduate student, Saint Petersburg State University; e-mail: kkqqkq77@outlook.com, phone: +7(963)249-76-01

Petrosyan Leon Aganesovich – professor, Saint Petersburg State University; e-mail: l.petrosyan@spbu.ru, phone: +7(812)363-65-37

$1, 2, \ldots, M\}$, where M is finite. As the number of players grows, M increases rapidly. When each player $i \in N$ is fully connected ($k_i = n$), only one graph exists ($M = 1$).

For a given network $G \in \mathbf{G}$, the set of connections of player i is defined as: $K_G(i) = \{j \mid (i,j) \in E(G)\}$, where $E(G)$ is the set of edges in graph G.

The state dynamics of the game is given by

$$\dot{x}^i(\tau) = f^i(x^i(\tau), u^i(\tau)), \quad x^i(t_0) = x_0^i \tag{1}$$

for $\tau \in [t_0, T]$ and $i \in N$. For notational convenience, we use $x(t)$ to denote the vector $(x^1(t), x^2(t), \ldots, x^n(t))$. We consider a special case where the payoff of the player i depends on his state variable and the state variables of players from the set $K(i)$. Thus, if the connections remain valued, the payoff of player i on the time interval $[t_0, T]$ is equal to

$$H_i = \sum_{j \in K_G(i)} \int_{t_0}^{T} h_i^j(x^i(\tau), x^j(\tau)) d\tau. \tag{2}$$

Here $h_i^j(x^i(\tau), x^j(\tau))$ is instantaneous gain through communication with player j, $i \in N$, $j \in K_G(i)$. We suppose that the term $h_i^j(x^i(\tau), x^j(\tau))$ is non-negative.

Let $\bar{x}(t) = (\bar{x}^1(t), \bar{x}^2(t), \ldots, \bar{x}^n(t))$ be the optimal cooperative trajectory. Denote by $\bar{u}(t) = (\bar{u}^1(t), \ldots, \bar{u}^i(t), \ldots, \bar{u}^n(t))$ the corresponding optimal cooperative control. Subject to the dynamics defined in [1], we suppose that players can cooperate to achieve the maximum joint payoff

$$\max_{u^1, u^2, \ldots, u^n} \sum_{i \in N} \sum_{j \in K_G(i)} \int_{t_0}^{T} h_i^j(x^i(\tau), x^j(\tau)) d\tau$$

$$= \sum_{i \in N} \left(\sum_{j \in K_G(i)} \int_{t_0}^{T} h_i^j(\bar{x}^i(\tau), \bar{x}^j(\tau))\, d\tau \right). \tag{3}$$

For simplicity in notation, we denote $\beta_{ij}(t) = \beta_{ij}(\bar{x}(t), T - t)$.

Let

$$\beta(x_0, T - t_0) = \{\beta_{ij}\}_{i,j=N}. \tag{4}$$

Here β is a weighted matrix of payoffs that players can get during the time $T - t_0$ by cooperation.

3. Dynamic network game with changing communication structure. The value characteristic function in the network $\mathbf{G}$ for the set N is equal to

$$V_G(x_0, T - t_0; N) = \sum_{i \in N} \left(\sum_{j \in K_G(i)} \beta_{ij}(x_0, T - t_0) \right). \tag{5}$$

We assumed that players choose such a network $\bar{G}$, which gives the maximum total payoff of players from the set N, i.e. $V_{\bar{G}}(x_0, T - t_0; N)$. We will call such a network $\bar{G}$ *optimal network*. Then the maximal payoff on the time interval $[t_0, T]$ is equal to

$$V(x_0, T - t_0; N) = \max_{G_m \in \mathbf{G}} V_G(x_0, T - t_0; N) = V_{\bar{G}}(x_0, T - t_0; N). \tag{6}$$

To determine the optimal network structure for each time interval, we solve the following optimization problem on each subinterval $[t_k, t_{k+1}]$:

$$\bar{V}^k(\bar{x}(t_k), t_{k+1} - t_k; N) = \max_{\mathbf{G}} V_G^k(\bar{x}(t_k), t_{k+1} - t_k; N), \tag{7}$$

- $\bar{V}^k$ represents the optimal joint cooperative payoff on the time interval $[t_k, t_{k+1}]$,
- $\mathbf{G}$ is the set of all possible networks,
- V_G^k denotes the payoff achievable under network $\mathbf{G}$.

Then the total payoff over the entire time horizon $[t_0, T]$ is computed as: $T_1 = \sum_{k=0}^{n-1} \bar{V}^k$.

For the time interval $[t_k, T]$, the optimization process begins at time t_0, where the total payoff over the entire interval $[t_0, T]$ is maximized, as defined by equation (6). The game is evaluated on the subinterval $[t_0, t_k)$ along the optimal trajectory $\bar{x}^\star(t)$, where $t \in [t_0, t_k)$, under the optimal network $\bar{G}^\star$. The resulting optimal payoff for this subinterval is denoted as: $V(x_0, t_k - t_0; N)$.

At time t_k, the objective shifts to identifying the optimal network $\bar{G}$ that maximizes the total payoff for all players over the remaining interval $[t_k, T]$. Denote the corresponding trajectory in this case by $\bar{\bar{x}}(t) = (\bar{\bar{x}}^1(t), \ldots, \bar{\bar{x}}^n(t))$.This is formulated as:

$$V(\bar{\bar{x}}(t_k), T - t_k; N) = \max_{G_m \in \mathbf{G}} V_G(\bar{\bar{x}}(t_k), T - t_k; N)$$

In this approval, the total payoff of the whole time interval will be $T_2 = V(\bar{\bar{x}}(t_k), T-t_k; N)+V(x_0, t_k-t_0; N)$. Here we can compare different total payoffs on $[t_k, T]$, $k = 1, 2, \ldots, n-1$, to identify that what is the best time point to change.

4. Numerical Example. Now we set the number of players is $N = 4$. And the limitation k_i of each node is 2 for $\forall i$. Firstly, we set up the weighted matrix $\beta(x_0, T-t_0)$ (see (5)), $W = \beta(x_0, T-t_0)$ for convenience. And we divide it into 5 subweighted matrix $W_k = \beta(x(t_k), t_{k+1}-t_k)$ on $[t_0, t_1], [t_1, t_2], \ldots, [t_4, T]$:

$$W = \begin{pmatrix} 0 & 7 & 8 & 8 \\ 7 & 0 & 5 & 7 \\ 8 & 5 & 0 & 10 \\ 8 & 7 & 10 & 0 \end{pmatrix}, \quad W_1 = \begin{pmatrix} 0 & 1 & 1 & 2 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 2 \\ 2 & 1 & 2 & 0 \end{pmatrix}, \quad W_2 = \begin{pmatrix} 0 & 2 & 5 & 2 \\ 2 & 0 & 1 & 2 \\ 1 & 5 & 0 & 3 \\ 2 & 2 & 3 & 0 \end{pmatrix},$$

$$W_3 = \begin{pmatrix} 0 & 1 & 1 & 2 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 2 \\ 2 & 1 & 2 & 0 \end{pmatrix}, \quad W_4 = \begin{pmatrix} 0 & 2 & 1 & 2 \\ 2 & 0 & 1 & 2 \\ 1 & 1 & 0 & 2 \\ 2 & 2 & 2 & 0 \end{pmatrix}, \quad W_5 = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}.$$

Then from the limitations we can get all possible networks (see fig. 1).

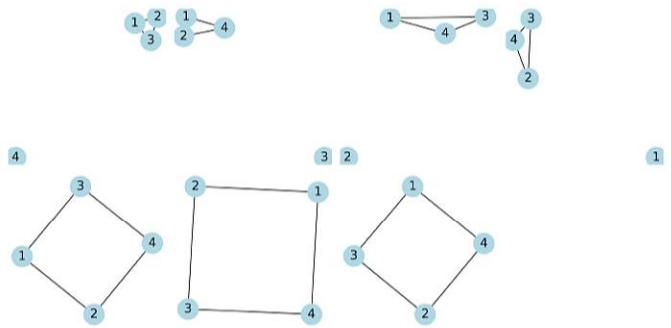


Figure 1. Possible networks

Based on the above analytics, in three distinct cases, we obtain the final results as follows:

1. For the whole time interval $[t_0, T]$, the optimal payoff is **32**, with the corresponding optimal network identified as **Graph 5**.

2. Under the five-part partitioning of $[t_k, T]$, the maximum total payoff increases to **33**, when we start changing our network structure to **Graph 6** at point t_2, and then transform the network structure to **Graph 4**, but this transforming didn't change the total payoff, so finally it remains **33** until the end.

3. For the k-th part division from $[t_0, t_1], \ldots, [t_{k-1}, T]$, the total payoff increases marginally to **35**.

5. Conclusion. The findings underscore the trade-offs inherent in network optimization over partitioned time intervals. While finer divisions enable improved total payoffs by tailoring network structures to evolving conditions, the added complexity in implementation and computation must be weighed against these gains.

References

1. Petrosyan L. A., Pankratova Y. B., Kong Q. About the instability of cooperative communication structures in differential network games // Contributions to Game Theory and Management, 2025. Vol. 17. P. 155–163.

UDC 519.832.3

Pu Y., Vostrenkova E. M.

A framework integrating environmental feedback and agent opinions. Evolutionary game theory and reinforcement learning

Recommended for publication by associate professor Gubar E. A.

1. Introduction. In this paper we consider an evolutionary system with environmental feedback and agent opinions. The main contribution of this paper is the combination of two different approaches. The standard solution of the system of differential equations for the evolution dynamics and some algorithms from reinforcement learning. These two approaches could be compared because the evolution game shows the adaptation of the population to the environment. The fundamental principle of reinforcement learning is the same.

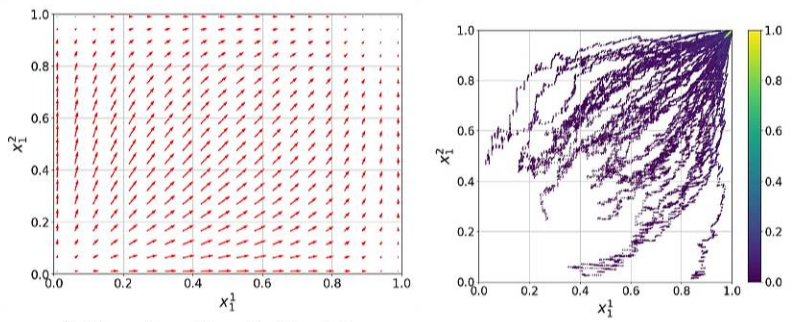
2. Relation between two theories. Both evolutionary game theory (EGT) and reinforcement learning theory (RLT) function as adaptive learning paradigms in dynamic environments, iteratively optimizing strategies through feedback mechanisms, ultimately driving the system toward equilibrium or optimal solutions [1]. More specifically, both frameworks drive strategy updates through performance differentials. In evolutionary games, fitness differentials parallel the advantage function in reinforcement learning, guiding strategies toward higher rewards. Furthermore, the replicator dynamics in evolutionary games share mathematical similarities with policy gradient updates in reinforcement learning, both embodying the "survival of the fittest" dynamic process.

For instance, obtain the direction field diagram of the Prisoner's Dilemma game by using Q-learning and replicator dynamics [2]. x_i^j is the probability for player j use strategy i. For Prisoners Dilemma first strategy is to cooperate and second strategy is to defect. Upon comparing the experimental results (Figure 1), it was found that the direction

Pu Yuxi – graduate student, Saint Petersburg State University; e-mail: st122375@student.spbu.ru, phone: +7(963)318-66-71

Vostrenkova Ekaterina Mikhailovna – graduate student, Saint Petersburg State University; e-mail: st077304@student.spbu.ru, phone: +7(952)237-91-92

field is similar to the replicator dynamics. This initially reveals the connection between reinforcement learning and evolutionary game theory at the model level.



a) The direction field of the replicator dynamic

b) The paths induced by the learning process

Figure 1. Comparison of direction field

However, in multi-agent systems, the environment often violates the Markov property, rendering the convergence and optimality guarantees of traditional RL invalid [3]. In contrast, evolutionary game theory, developed from fields such as biological evolution and economics, relaxes assumptions like the requirement for perfect rationality in traditional game theory. This framework better aligns with the behavioral characteristics of real-world agents and provides a theoretical foundation for analyzing multi-agent decision-making and interactions in complex environments [4, 5].

3. Model. In this study, we consider a large but finite population inhabiting a finite space. The state of the population is assumed to change as a result of random pairwise interactions between its agents [6]. The population can be modeled as consisting of two types of agents or behaviors, which we call strategies. Furthermore, we assume that each agent has a binary opinion about the environment regardless of its strategy. That is, the first opinion is that "there are a lot of resources in the environment" and the second opinion is that "there are a little of resources in the environment". By environment we mean the resources available to the agents.

An evolutionary game with environment-opinion feedback can be represented as [7, 8]

$$\begin{cases} \dot{x} = x(1-x)(u(e^1, x_N, A_y) - u(e^2, x_N, A_y)), \\ \dot{n} = n(1-n)(\theta x + \psi(1-x)), \\ \dot{y} = (1-y)p_{21} - yp_{12}, \end{cases} \tag{1}$$

where x is the fraction of the population using the first pure strategy, x_N is the state of the population, e^i is the vector corresponding to the i-th net strategy of the player, y is the proportion of agents in the population holding first opinion, $A_y = yA_1 + (1-y)A_0$ is payoffs matrix, $u(e^i, x_N, A) = e^i A x_N$ is the expected payoff of the agent with pure strategy i when facing a random opponent. The state of the environment is described by the parameter $n(t)$, $n \in [0, 1]$, where $n = 0$ ($n = 1$) when the environment is completely consumed (replenished) [9]. Agents following the first pure strategy replenish resources n at a rate $\theta > 0$, while agents following the second pure strategy cause resources to be depleted at a rate $\psi = -1$. The imitation protocol is [10]

$$p_{ij} = \left[y_j \sum_{l=1}^{2} x_l u(e^l, x_N, A_n) b_{jl} - y_i \sum_{l=1}^{2} x_l u(e^l, x_N, A_n) b_{il} \right]_0^1,$$

where b_{ij} represent the degree of confidence of the agent with opinion m_i in the agent pursuing strategy j, $A_n = nA_1 + (1-n)A_0$ is payoff matrix.

4. Numerical experiment. The Prisoner's Dilemma. Let's briefly discuss two algorithms used in the study. Exact algorithm solves a system of differential equations using a method built into Matlab. It is based on an explicit Runge–Kutta method, namely the Dormand–Prince pair of orders 5 and 4 (Matlab `ode45` procedure). The idea of another algorithm is to interpret Q-value as Boltzmann probability of action selection, construct continuous time limit of Q-learning model, introduce active control mechanism through Q-learning, and explain the exploration-utilization mechanism of reinforcement learning in combination with replication dynamics.

A series of experiments for the Prisoner's Dilemma is considered in this study. In the current experiment, the Prisoner's Dilemma is selected as the base game for its economic interpretation, which describes the

interaction of agents. In this game, the first strategy corresponds to the player's choice to cooperate, and the second to the choice to defend. We introduce two player payoff matrices

$$A_0 = \begin{pmatrix} 3.5 & 1 \\ 2 & 0.75 \end{pmatrix}, \quad A_1 = \begin{pmatrix} 4 & 1 \\ 4.5 & 1.25 \end{pmatrix}, \tag{2}$$

for which the following relations are true:

$$a_{11}^0 > a_{21}^0,\ a_{12}^0 > a_{22}^0, \quad a_{11}^1 < a_{21}^1,\ a_{12}^1 < a_{22}^1. \tag{3}$$

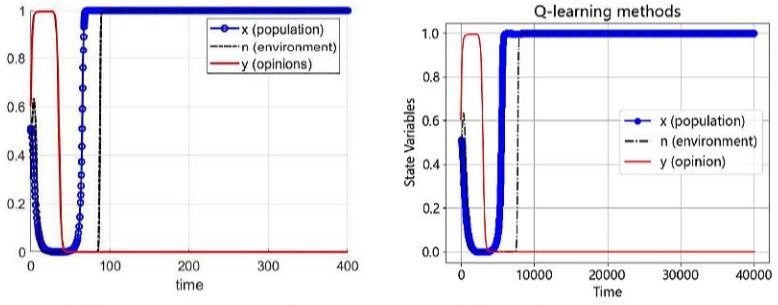
It is evident in the Prisoner's Dilemma game that there exists a single Nash equilibrium state, contingent on the relations (3). For the specific game delineated by the matrix A_1 (A_0), this state is characterised as strategy profile (e^2, e^2) (or (e^1, e^1)).

In the present numerical experiment, the parameters of the system (1) are specified as follows:

Table 1. System parameters

Param.	Value	Description
θ	2	The rate of replenishment of resources by cooperators
ψ	-1	The rate at which defectors consume resources
x_0	0.5	Proportion of cooperators in the population at the initial time moment
y_0	0.6	Proportion of agents with an opinion m_1 at initial time moment
n_0	0.3	Environmental condition at initial time moment
B	$\begin{pmatrix} 0.999 & 0.001 \\ 0.001 & 0.999 \end{pmatrix}$	Agents with opinion m_1 (m_2) are assumed to trust only cooperators (defenders)

We present the results of the numerical simulation



a) Standard approach b) Reinforcement Learning

Figure 2. Comparison of experimental results

As illustrated in the Figure 2, both algorithms result in the same stationary point. Moreover, the results from both algorithms are the same for all series of numerical experiments. Therefore, the RL algorithm could be used to solve more complex evolutionary systems.

References

1. Fuente N., Alonso M., Casadell G. Game theory and multi-agent reinforcement learning: from Nash equilibria to evolutionary dynamics // arXiv preprint. arXiv: 2412.20523, 2024.
2. Dolgopolov A. Reinforcement learning in a prisoner's dilemma // Games and Economic Behavior. 2024. Vol. 144. P. 84–103.
3. Tembine H. Deep Learning meets Game Theory. 2015.
4. Tuyls K., Verbeeck K., Lenaerts T. A selection-mutation model for q-learning in multi-agent systems // The Second International Joint Conference on Autonomous Agents & Multiagent Systems, Proceedingss. 2003. P. 693–700.
5. Tuyls K., Nowe A. Evolutionary game theory and multi-agent reinforcement learning // The Knowledge Engineering Review. 2005. Vol. 20. P. 63–90.
6. Sandholm W. H. Population Games and Evolutionary Dynamics. Cambridge: MIT Press, 2010. 616 p.
7. Lorits E. M., Gubar E. A. An evolutionary game with environmental feedback and players' opinions // Large-Scale Systems Control. Vol. 106. M.: Trapeznikov Institute of Control Sciences of RAS. 2023. P. 172–183. (in Russian)
8. Lorits E. M. Evolutionary model with environmental feedback and agent opinions // Control Processes and Stability. 2024. Vol. 11. No 1. P. 355–360.
9. Weitz J. S., Eksin C., Paarporn K. et al. An oscillating tragedy of the commons in replicator dynamics with game-environment feedback // PNAS. 2016. Vol. 113. No 47. P. E7518–E7525.
10. Riehl J. R., Cao M. Control of stochastic evolutionary games on networks // IFAC. 2015. Vol. 48. Iss. 22. P. 76–81.

UDC 510

Qu R., Huang R.

Integrating Raft algorithm to blockchain consensus mechanism

1. Research background and technical introduction. With blockchains development, its decentralized, tamper-proof, and traceable features have been widely applied in finance, supply chains, and government [1]. Consensus mechanisms,as core components, directly affect performance and scalability. Mainstream mechanisms like PoW and PBFT have high energy use and low efficiency, unsuitable for small-scale or high-frequency scenarios [2]. Raft is known for its clear structure, simplicity, high efficiency, scalability, and fault tolerance [3]. This paper studies Rafts integration and optimization in blockchain, using agricultural traceability as an example.

2. Modeling process. In Table 1 we define the denotations we use in the model.

Table 1. Denotations

Symbol	Meaning
T_{total}	Total latency (block final confirmation time)
s	Size of each transaction (unit: bytes)
λ	Transactions per second (TPS)
B	Block size (unit: bytes)
d	One-way network transmission delay (unit: seconds)
$\varepsilon(N)$	Consensus delay overhead, related to the number of nodes N (e.g., voting, synchronization time)
BW_i	Bandwidth of the i-th node (unit: bytes/second)
L_i	Additional delay of the i-th node (queuing, processing, I/O, etc.)
$\max\{\dots\}$	Takes the transmission + processing time of the slowest node to avoid system bottleneck

We compute the times that make up the total delay as follows.

1. Transaction packaging time

$$T_{\text{pack}} = \frac{s\lambda}{B}. \tag{1}$$

Qu Renhao – PhD, Yan'an University; e-mail: 2097207990@qq.com, phone: +8613377970197

Huang Ruiguang – PhD, Yan'an University; e-mail: 1938942010@qq.com, phone: +8618694009662

2. Network transmission delay

$$T_{\text{net}} = 2d. \tag{2}$$

3. Consensus processing delay:

$$T_{\text{consensus}} = \varepsilon(N). \tag{3}$$

4. Bottlenecks in node processing time:

$$\max\left\{\frac{s}{BW_i} + L_i\right\}. \tag{4}$$

The total delay thus is

$$T_{\text{total}} = \frac{s\lambda}{B} + 2d + \varepsilon(N) + \max\left\{\frac{s}{BW_i} + L_i\right\}. \tag{5}$$

We have chosen for the experiment the parameter values presented in Table 2.

Table 2. Parameter Settings

Parameter	Value	Description
s	2 KB	Size of one transaction (farmer upload data)
λ	100 TPS	System throughput (transactions per second)
B	200 KB	Maximum block size
d	0.05 s	One-way network latency
$\varepsilon(3)$	0.15 s	Raft consensus synchronization delay (3 nodes)
BW_i	500 KB/s	Bandwidth of the slowest node
L_i	0.1 s	Node processing delay (disk/cache, etc.)

By substituting the values, we can estimate the maximum delay for the complete confirmation and on-chain process of each transaction:

$$T_{\text{total}} = \frac{2048 \times 100}{204800} + 2 \times 0.05 + 0.15 + \max\left\{\frac{2048}{512000} + 0.1\right\} = 1.354 \text{ s}.$$

3. Optimization. Here we try to optimize the performance by varying the parameters.

3.1. Max delay vs. block size and TPS. Under various combinations of block size B and transaction throughput λ, the system's estimated maximum delay T_{total} can be calculated (Fig. 1).

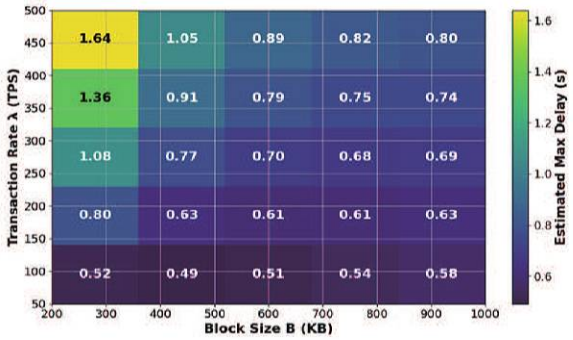


Figure 1. Max delay vs. block size and TPS

Optimization suggestions. Increasing block size reduces delay. At the same time lowering transaction rate eases system pressure. The lower-right area (large block size + moderate TPS) is optimal, though for really low TPS smaller block size might work better.

3.2. Max delay vs. block size for TPS = 100. Under the assumption of TPS = 100, we simulated the impact of different numbers of nodes N on the maximum system delay, as shown in Figure 2.

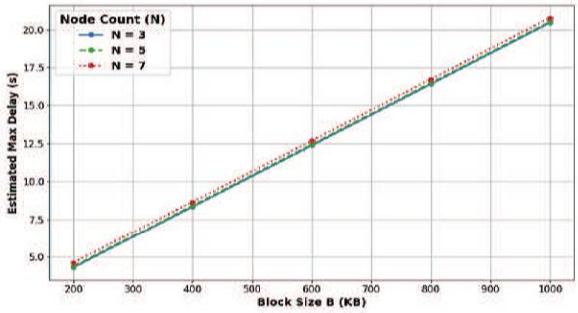


Figure 2. Max delay vs. block size (TPS = 100)

Conclusion. Delay increases linearly with the increase in block size. Since larger blocks take longer to fill and longer to transmit over the network, the total delay increases. The more nodes there are, the higher is the consensus delay.

3.3. Max delay vs. bandwidth and I/O delay. Under fixed node count $N = 3$, fixed block size, and TPS, the impact of the slowest node's bandwidth BW_i and I/O delay L_i on the maximum system delay is presented in Figure 3.

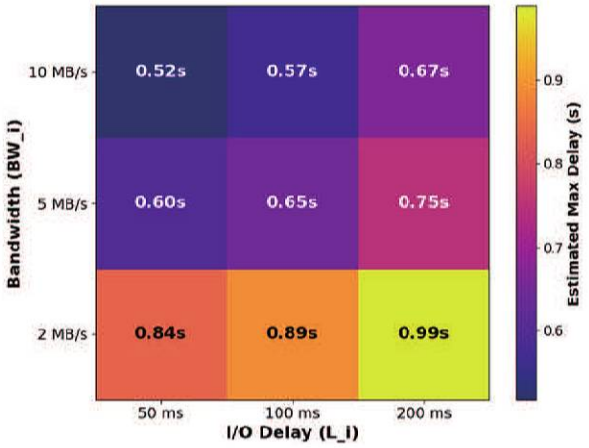


Figure 3. Max delay vs. bandwidth and I/O delay

3.4. Optimization conclusions. Higher bandwidth and lower I/O delay improve performance; the top-left region (high bandwidth, low delay) performs best, while the bottom-right (low bandwidth, high delay) is worst. Transaction per second (TPS) and total latency are two major indicators for measuring the performance of blockchain traceability systems. To achieve a balanced optimization of both, the following conclusions can be drawn based on the previous.

1. Transaction packaging time (1) can be reduced by adjusting the block size B:
$$B_{\text{opt}} > B \implies \frac{s\lambda}{B_{\text{opt}}} < T_{\text{pack}}.$$
2. Optimizing the network environment d decreases network transmission latency (2):
$$d_{\text{opt}} < d \implies 2d_{\text{opt}} < T_{\text{net}}.$$
3. The consensus overhead (3) should be reduced $\varepsilon^{\text{opt}}(N) < \varepsilon(N)$.

4. Increase in node bandwidth and processing speed reduces node bottleneck processing time (4):

$$BW_i^{\text{opt}} > BW_i,\ L_i^{\text{opt}} < L_i \implies \max_i \left(\frac{s}{BW_i^{\text{opt}}} + L_i^{\text{opt}} \right) < \max_i \left(\frac{s}{BW_i} + L_i \right).$$

Overall post-optimization total latency is

$$T_{\text{total}}^{\text{opt}} = \frac{s\lambda}{B_{\text{opt}}} + 2d_{\text{opt}} + \varepsilon^{\text{opt}}(N) + \max_i \left(\frac{s}{BW_i^{\text{opt}}} + L_i^{\text{opt}} \right).$$

4. Conclusion. The analysis shows that optimizing block size, node count, bandwidth, and I/O latency in blockchain systems can reduce transaction delay and boost performance. Higher bandwidth and lower I/O latency significantly improve efficiency, while increasing block size and node count involves balancing throughput and latency.

References

1. Panigrahi A., Pati A., Dash B., et al. ASBlock: An agricultural based supply chain management using blockchain technology // Procedia Computer Science. 2024. Vol. 235. P. 1943–1952.

2. Mokgomola F., Telukdarie A., Munien I., et al. Blockchain and smart contracts based agricultural supply chain // Procedia Computer Science. 2024. Vol. 237. P. 637–644.

3. Sun W., Bai X., Shi B., et al. TD-Raft: A consensus algorithm with inspection mechanism and server performance threshold // IAENG International Journal of Computer Science. 2025. Vol. 52. No 4. P. 1070–1076.

UDC 519.83

Sun S., Petrosyan L. A.

Dynamic hierarchical games with changing communication structure and pairwise interactions

1. Introduction. In [1] it has been shown that the cooperative network can change over time, due to the time inconsistency of cooperative communication structure. Recent studies have significantly advanced the modeling of complex problems using differential games on networks [2,3]. We consider a two-level hierarchical system [4] in a differential network game defined on a time interval $[t_0, T]$, and illustrate how the cooperative communication structure changes at a specified time instant along the trajectory maximizing the joint players payoff.

2. Cooperative communication in dynamic hierarchical games with pairwise interactions. Consider a two-level hierarchical system on a finite game horizon $[t_0, T]$, consisting of two control centers, A_i,$(i = 1, 2)$, and n subordinate departments B_j $(j = 1, \ldots, n)$. Denote the set of all players by $N = \{A_1, A_2, B_1, \ldots, B_n\}$, and the set of all arcs $L = \{\text{arc}(A_i, B_j) | A_i \in N, B_j \in N\}$. Define the set of B_j connected to A_i as $K_i = \{B_j : \text{arc}(A_i, B_j) \in L\}$, $A_i, B_j \in N$. Similarly, $\widetilde{K}_j = \{A_i : \text{arc}(A_i, B_j) \in L\}$. Suppose that $\widetilde{K}_j$ satisfies the constraints for a given fixed set $M \subseteq N$: $|\widetilde{K}_j| = 1, B_j \in M \subseteq N; |\widetilde{K}_j| = 2, B_j \notin M$.

Let $x^{ij}(\tau) \in \mathbb{R}^m$, $u^{ij}(\tau) \in U^{ij} \subset \mathbb{R}^k$ represent the states and control variables of players $A_i \in N$ interacting with $B_j \in K_i$ at time τ. Similarly, define $\widetilde{x}^{ji}(\tau) \in \mathbb{R}^m$ $\widetilde{u}^{ji}(\tau) \in \widetilde{U}^{ji} \subset \mathbb{R}^k$ for players $B_j \in N$ interacting with A_i. U^{ij}, $\widetilde{U}^{ij}$ are compact sets. The state dynamics is:

$$\dot{x}^{ij}(\tau) = f^{ij}(x^{ij}(\tau), u^{ij}(\tau)), x^{ij}(t_0) = x_0^{ij},\ A_i \in N, B_j \in K_i. \qquad (1)$$

$$\dot{\widetilde{x}}^{ji}(\tau) = \underline{f}^{ji}(\widetilde{x}^{ji}(\tau), \widetilde{u}^{ji}(\tau)), \widetilde{x}^{ji}(t_0) = \widetilde{x}_0^{ji},\ B_j \in N, A_i \in \widetilde{K}_j. \qquad (2)$$

where $\tau \in [t_0, T]$, $f^{ij}(x^{ij}(\tau), u^{ij}(\tau))$,$\underline{f}^{ji}(\widetilde{x}^{ji}(\tau), \widetilde{u}^{ji}(\tau))$ are assumed to be continuously differentiable in $x^{ij}(\tau)$, $u^{ij}(\tau)$; $\widetilde{x}^{ji}$,$\widetilde{u}^{ji}(\tau)$. Define the payoffs of players A_i: $H_i(x^i, \widetilde{x}^j, T - t_0) = \sum_{B_j \in K(i)} \int_{t_0}^{T} h_i^j(x^{ij}(\tau), \widetilde{x}^{ji}(\tau))\, d\tau$

Sun Shulei – graduate student, Saint Petersburg State University; e-mail: st122369@student.spbu.ru, phone: +7(963)249-46-38

Petrosyan Leon Aganesovich – professor, Saint Petersburg State University; e-mail: l.petrosyan@spbu.ru, phone: +7(812)363-65-37

and of B_j: $\widetilde{H}_j(\widetilde{x}^j, x^i, T-t_0) = \sum_{A_i \in \widetilde{K}(j)} \int_{t_0}^{T} \widetilde{h}_j^i(\widetilde{x}^{ji}(\tau), x^{ij}(\tau))\, d\tau$. We assume that $h_i^j(x^{ij}(\tau), \widetilde{x}^{ji}(\tau))$, $\widetilde{h}_j^i(\widetilde{x}^{ji}(\tau), x^{ij}(\tau))$, for $A_i, B_j \in N$ are non-negative. For simplicity, define the maximal joint payoff of player A_i and B_j on $\mathrm{arc}(A_i, B_j)$ as:

$$\beta_{ij}(\bar{x}(t), T-t) = \int_t^T (h_i^j(\bar{x}^{ij}(\tau), \bar{\widetilde{x}}^{ji}(\tau)) + \widetilde{h}_j^i(\bar{\widetilde{x}}^{ji}(\tau), \bar{x}^{ij}(\tau)))\, d\tau. \quad (3)$$

Denote the optimal cooperative trajectory maximizing (4) subject to (1) and (2) on graph G by $\bar{x}(t) = (\bar{x}^1(t), \bar{x}^2(t), \bar{\widetilde{x}}^1(t) \ldots, \bar{\widetilde{x}}^n(t))$, $\bar{x}^i(t) = (\bar{x}^{ij}(t), B_j \in K(i))$, $\bar{\widetilde{x}}^j(t) = (\bar{\widetilde{x}}^{ji}(t), A_i \in \widetilde{K}(j))$, $t \in [t_0, T]$:

$$V_G(N, \bar{x}, T-t) = \max_{u^i, \widetilde{u}^j : A_i, B_j \in N} \Bigg(\sum_{A_i \in N} \sum_{B_j \in K(i)} \int_t^T h_i^j(x^{ij}(\tau), \widetilde{x}^{ji}(\tau))\, d\tau + \sum_{B_j \in N} \sum_{A_i \in \widetilde{K}(j)} \int_t^T \widetilde{h}_j^i(\widetilde{x}^{ji}(\tau), x^{ij}(\tau))\, d\tau \Bigg). \quad (4)$$

3. Dynamic hierarchical network with pairwise interaction and changing communication structure. Now, we shall consider the optimal hierarchical network with a variable communication structure. We firstly denote the adjacency matrix of the graph G on time interval $[t_0, T]$ by δ such that

$$\delta_{ij} = \begin{cases} 1, & \text{if } \mathrm{arc}(A_i, B_j) \text{ exists in graph } G, \\ 0, & \text{else.} \end{cases}$$

As mentioned before, $|\widetilde{K}_j| = 1$ if $B_j \in M \subseteq N$ and $|\widetilde{K}_j| = 2$ if $B_j \notin M$. Thus δ_{ij} satisfy the conditions:

$$\sum_{i=1}^{2} \delta_{ij} = 1 \text{ if } j \in M, \quad \sum_{i=1}^{2} \delta_{ij} = 2 \text{ if } j \in N \setminus M. \quad (5)$$

We now consider two cases. At first, we maximize the joint cooperative payoff on a fixed network. Then, at time instant $t_1 = t_0 + (T-t_0)/2$, we allow for the possibility of changing the network.

The objective of the first case is to obtain the maximized joint cooperative payoff on the entire time interval $[t_0, T]$ with a single optimal

network G^*. The adjacency matrix of G^* is denoted by $\delta^* = (\delta^*_{ij}), A_i$, $B_j \in N$. In this case, the maximal joint cooperative payoff, designated as V_1 is calculated as follows:

$$V_1(N, x_0, T - t_0, \delta^*) = \max_{\delta_{ij}, \forall A_i, B_j \in N} \sum_{A_i \in N} \sum_{B_j \in N} [\delta_{ij} \cdot \beta_{ij}(x_0, T - t_0)]. \quad (6)$$

Denote the optimal cooperative trajectory in this case by $\bar{x}^*(t) = (\bar{x}^{1*}(t), \bar{x}^{2*}(t), \bar{\bar{x}}^{1*}(t), \ldots, \bar{\bar{x}}^{n*}(t))$, $\bar{x}^{i*}(t) = \left(\bar{x}^{ij}(t),\ B_j \in \{B_j \mid \delta^*_{ij} = 1\}\right)$ $\bar{\bar{x}}^{j*}(t) = \left(\bar{\bar{x}}^{ji}(t),\ A_i \in \{A_i \mid \delta^*_{ij} = 1\}\right)$, where $\bar{x}^{ij}(t)$, $\bar{\bar{x}}^{ji}(t)$ are the optimal trajectories that maximize the joint payoffs of players A_i, B_j on arc (A_i, B_j) according to (6), for $A_i, B_j \in N$, $t \in [t_0, T]$.

In the second case, we assume that players have the possibility to change network by modifying the pairwise interactions at time instant $t_1 = t_0 + \frac{T - t_0}{2}$.

At time instant t_0, we maximize the total payoff on time interval $[t_0, T]$, as defined by equation (6) , then the game evaluates on the time interval $[t_0, t_1)$ along the corresponding trajectory $\bar{x}^*(t)$,$t \in [t_0, t_1)$, with the network G^*, the total payoff on this time interval is obtained:

$$V_2(N, x_0, t_1 - t_0, \delta^*) = \sum_{A_i \in N} \sum_{B_j \in N} \delta^*_{ij} \cdot \beta_{ij}(x_0, t_1 - t_0). \quad (7)$$

At time instant t_1, the objective is to identify the graph $\bar{G}$, with the adjacency matrix $\bar{\delta}$, that maximizes the total payoff of all players over the time interval $[t_1, T]$. Notice that there are some arcs $(A_{i'}, B_{j'})$ which did not exist in the graph G^*. Consequently, it is necessary to construct a subgame within the time interval $[t_1, T]$ to define the optimal joint payoffs $\alpha'_{i'j'}(\bar{\bar{x}}(t_1), T - t_1)$ in all the arcs $(A_{i'}, B_{j'}) \notin G^*$:

$$\begin{aligned} \alpha'_{i'j'}(\bar{\bar{x}}(t_1), T - t_1) = &\int_{t_1}^{T} h^{j'}_{i'}(\bar{x}^{i'j'}(\tau - (t_1 - t_0)), \bar{\bar{x}}^{j'i'}(\tau - (t_1 - t_0)))\, d\tau \\ &+ \int_{t_1}^{T} \widetilde{h}^{i'}_{j'}(\bar{\bar{x}}^{j'i'}(\tau - (t_1 - t_0)), \bar{x}^{i'j'}(\tau - (t_1 - t_0)))\, d\tau. \end{aligned} \quad (8)$$

Subsequently, in the subgame over time interval $[t_1, T]$, the optimal joint payoffs $\alpha_{ij}(\bar{x}(t_1), T - t_1)$ on all arcs $(A_{i'}, B_{j'}) \in N$ can be defined as:

$$\alpha_{ij}(\bar{\bar{x}}(t_1), T - t_1) = \begin{cases} \alpha'_{ij}(\bar{\bar{x}}(t_1), T - t_1) & \text{if } \operatorname{arc}(A_i, B_j) \notin G^*, \\ \beta_{ij}(\bar{x}(t_1), T - t_1) & \text{if } \operatorname{arc}(A_i, B_j) \in G^*. \end{cases} \quad (9)$$

Then we get the network $\bar{G}$ corresponding to (10), notice that $\bar{G}$ may be the same or different from the previous network G^*.

$$V_2(N, \bar{\bar{x}}(t_1), T-t_1, \bar{\delta}) = \max_{\substack{\delta_{ij}, \\ \forall A_i, B_j \in N}} \sum_{A_i \in G} \sum_{B_j \in N} \delta_{ij} \cdot \alpha_{ij}(\bar{\bar{x}}(t_1), T-t_1). \quad (10)$$

Then move on time interval $[t_1, T]$ along the corresponding optimal trajectory without changing the network. Denote the optimal cooperative trajectory in this case by $\bar{\bar{x}}(t) = \left(\bar{\bar{x}}^1(t), \bar{\bar{x}}^2(t), \bar{\bar{\bar{x}}}^1(t), \ldots, \bar{\bar{\bar{x}}}^n(t)\right)$. As a result, we can get the total payoff denoted by V_2, and if $V_2(N, \bar{\bar{x}}(t_1), T - t_1, \bar{\delta}) > \big(V_1(N, x_0, T-t_0, \delta^*) - V_1(N, x_0, t_1 - t_0, \delta^*)\big)$, we will change the network, then get the joint players' payoff $V_2 = V_2(N, x_0, t_1 - t_0, \delta^*) + V_2(N, \bar{\bar{x}}(t_1), T - t_1, \bar{\delta})$, or if it is impossible to increase the joint payoff by changing networks we shall not change at time instant t_1.

4. Example of hierarchical network with pairwise interaction and changing communication structure. Denote $N = \{A_1, A_2, B_1, ..., B_7\}$ the set of all players. Suppose the optimal trajectories on each pairwise intersections , which maximize the payoffs of the two linked players are $\bar{x}^{ij}(\mathrm{t})$ and $\bar{\bar{x}}^{ji}(\mathrm{t})$, for $t \in [t_0, T]$, $A_i, B_j \in N$. Now, we consider a numerical example by introducing matrices: $\beta(\bar{x}(t), T-t) = \big(\beta_{ij}(\bar{x}(t), T-t_0)\big)$, Suppose that:

$$\begin{aligned}
\beta(x_0, T-t_0) &= \begin{bmatrix} 19 & 33 & 33 & 25 & 31 & 32 & 17 \\ 26 & 29 & 29 & 29 & 27 & 32 & 20 \end{bmatrix} \\
&= \beta(x_0, t_1 - t_0) + \beta(\bar{x}(t_1), T - t_1) \\
&= \begin{bmatrix} 16 & 27 & 23 & 11 & 25 & 19 & 6 \\ 19 & 22 & 17 & 20 & 21 & 21 & 14 \end{bmatrix} + \begin{bmatrix} 3 & 6 & 10 & 14 & 6 & 13 & 11 \\ 7 & 7 & 12 & 9 & 6 & 11 & 4 \end{bmatrix},
\end{aligned}$$

where $t_1 = t_0 + \frac{T-t_0}{2}$. The connections in the network satisfy the following condition: $|\widetilde{K}_j| = 1$, for $j = 1, 2, 3$, and $|\widetilde{K}_j| = 2$, for $j = 4, \ldots, 7$.

At first, under the conditions (5) by (6),we obtain the fixed optimal network G^* with the adjacency matrix $\delta^* = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$, and the value $V_1 = 305$, on the entire time interval.

Subsequently, we shall consider if there is a possibility to modify the network at t_1. The network remains on $[t_0, t_1]$, therefore the cooperative payoff can be obtained during this interval using the network G^*. So we get $V_2(N, x_0, t_1 - t_0, \delta^*) = 206$ (see (7)).

At time instant $t = t_1$, with δ^*, it was observed that $\mathrm{arc}(A_1, B_1)$, $\mathrm{arc}(A_2, B_2), \mathrm{arc}(A_2, B_3)$ were not linked on the time interval $[t_0, t_1]$. Consequently, based on (8) and (9), $\alpha(\bar{\bar{x}}(t_1), T - t_1) = (\alpha_{ij}(\bar{\bar{x}}(t_1), T - t_1)) = \begin{bmatrix} 16 & 6 & 10 & 14 & 6 & 13 & 11 \\ 7 & 22 & 17 & 9 & 6 & 11 & 4 \end{bmatrix}$.

The optimal network $\bar{G}$ of the subgame has the adjacency matrix $\bar{\delta} = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$, maximizes the total payoff $V_2(N, \bar{\bar{x}}(t_1), T - t_1, \bar{\delta}) = 129$ on the time interval $(t_1, T]$

The total joint payoff resulting from the altered communicative structure throughout the entire time interval $[t_0, T]$ is $V_2(N, x_0, T - t_0, \delta^*, \bar{\delta}) = V_2(N, x_0, t_1 - t_0, \delta^*) + V_2(N, \bar{\bar{x}}(t_1), T - t_1, \bar{\delta}) = 335 > 305 = V_1$.

5. Conclusion. The example indicates that an increase in joint payoffs may be achieved by modifying the communication structure. However, in the absence of an increase in joint payoffs, it would not be feasible to implement changes to the network at the specified time instants.

References

1. Petrosyan L. A., Pankratova Y. B., Kong Q. About the instability of cooperative communication structures in differential network games // Contributions to Game Theory and Management, 2025. Vol. 17. P. 155–163.

2. Pai H. M. A differential game formulation of a controlled network // Queueing Systems. 2010. Vol. 64. P. 325–358.

3. Wie B. W. A differential game model of Nash equilibrium on a congested traffic network // Networks. 1993. Vol. 23. No 6. P. 557–565.

4. Petrosyan L. A., Zenkevich N. A. Game theory (Series on Optimization. Vol. 3). Singapore: World Scientific Publishing Co. Pte. Ltd. 1996. 364 p.

UDC 519.83

Xiong X.

A history-driven dynamic network game model for smart home systems: Comparative analysis with Q-learning

Recommended for publication by associate professor Balykina Y. E.

1. Introduction. Smart home systems act as cyber-physical systems (CPS) that integrate devices to optimize energy and automation. Dynamic changes challenge traditional methods such as Q-learning. In this paper, we propose a History-Driven Dynamic Network Game Model with Spanning Tree Optimization (HDNG-STO) for smart homes, comparing it with Q-learning to improve efficiency and adaptability. Our contribution is to apply HDNG-STO to device interaction, simulate its performance and compare it with Q-learning, and analyze the metrics in dynamic CPS. Subsequent sections include related work, modeling, baselines, results, and future directions.

2. Related work. Dynamic network optimization in CPS has been explored through game theory [1] and reinforcement learning [2]. Bergantios et al. [3] proposed game-theoretic approaches for minimum cost spanning trees, while Q-learning has been applied to adaptive control in smart homes [4,5]. However, Q-learning struggles with scalability and convergence in dynamic networks due to its static reward structures. The HDNG-STO model integrates historical data into strategy updates, offering a promising alternative for CPS with evolving topologies.

3. HDNG model for smart homes.

3.1. Model overview. In smart homes, devices (e.g., thermostat, lights) are players in a dynamic game, adjusting strategies u_j based on historical contributions H_j^t, optimized via minimum-cost spanning trees.

3.2. Network structure and players. Players adjust strategies $U_j = \{u_{j,k}\}$ (1 for communication, 0 otherwise) iteratively. Information $d(\overrightarrow{j_1, j_2})$ is transmitted when $u_{j_1,j_2} = u_{j_2,j_1} = 1$, limited by capacity

Xiong Xiao – graduate student, Saint Petersburg State University; e-mail: txxiongxiao@163.com, phone:+7(963)318-66-75

$C_{e_{j_1,j_2}}$. History H_j^t updates as:

$$H_j^t = \frac{\sum_{t'=0}^{t-1} \sum_{k \neq j} d^{t'}(\overrightarrow{j,k}) + d^{t'}(\overrightarrow{k,j})}{\sum_i \sum_{t'=0}^{t-1} \sum_{l \neq i} d^{t'}(\overrightarrow{i,l}) + d^{t'}(\overrightarrow{l,i})} \text{ for } t > 0.$$

Definition 1 *Define the total transmission amount d_j^t of player j in stage t:*

$$d_j^t = \sum_{j_2 \in V, j_2 \neq j} d^t(\overrightarrow{j,j_2}) + d^t(\overrightarrow{j_2,j}).$$

3.3. Dynamic game process. The process iterates until $|C^{t+1} - C^t| < 0.001$ or $\max_j |H_j^{t+1} - H_j^t| < 0.001$. Steps include:

- **Initialization:** Set U_j^0 and H_j^0.
- **Strategy adjustment:** Prioritize connections based on H_j^t.
- **Information transfer:** Update $d(\overrightarrow{j_1,j_2})$.
- **History update:**

$$H_j^{t+1} = H_j^t(1 - 0.03) + 0.03 \cdot \frac{d_j^t}{\max(D^t, 0.01)}.$$

- **Spanning tree optimization:** Use

$$C_{ij}^t = \min\left(\frac{0.5}{\max(H_i^t, 0.01) + \max(H_j^t, 0.01) + 0.03}, 10\right).$$

3.4. Termination. Record final T_{final} and H_j^{final} when stable.

4. Q-learning baseline. Q-learning learns optimal policies via:

$$Q(s,a) \leftarrow Q(s,a) + 0.1\left[r + 0.9 \max_{a'} Q(s',a') - Q(s,a)\right].$$

States are connectivity patterns, actions are binary communication decisions, and rewards are $r = -\sum_{j,k} d(\overrightarrow{j,k})$. An ϵ-greedy strategy uses $\epsilon = 0.7$ for $t \leq 50$, 0.4 for $50 < t \leq 100$, and 0.2 for $t > 100$.

4.1. Limitations. Q-learning lacks topology awareness, scales poorly, and adapts slowly to dynamics.

5. Simulation and results.

5.1. Setup. Simulate 5 devices over 200 iterations. Parameters:

- **HDNG-STO:** $\alpha = 0.03$, $\epsilon_C = 0.001$, $\epsilon_H = 0.001$, $c_D = 0.01$.
- **Q-Learning:** $\alpha = 0.1$, $\gamma = 0.9$, ϵ decays as described in Section .

Metrics: convergence time, energy cost, and stability (variance of H_j^t).

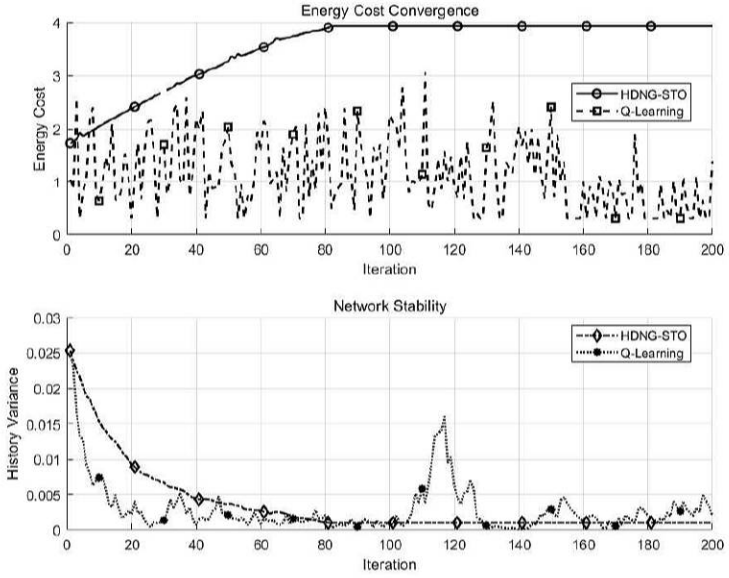


Figure 1. Simulation results comparing HDNG-STO and Q-Learning: (a) Energy Cost Convergence, (b) Network Stability

5.2. Results. Simulation results, based on the optimized MATLAB implementation, are visualized in Figure 1 and summarized as follows:

- **Convergence time:** HDNG-STO converges in 103 iterations, while Q-learning converges in 49 iterations.

- **Energy cost:** HDNG-STO achieves an average cost of 3.40, compared to 1.42 for Q-learning, indicating Q-learnings superior energy efficiency.
- **Stability:** HDNG-STO exhibits a history variance of 0.0039, compared to 0.0030 for Q-learning, suggesting HDNG-STOs greater network stability, though Q-learning shows more fluctuation in early iterations.

6. Conclusion and future work. HDNG-STO offers superior stability in smart home CPS via history-driven spanning tree optimization, while Q-learning excels in convergence and energy efficiency but struggles with stability. Future work includes enhancing HDNG-STOs efficiency, reducing Q-learnings variance, and testing scalability with stochastic elements.

References

1. Fudenberg D., Tirole J. Game theory. Cambridge: The MIT press, 1991. 603 p.
2. Skarding J., Gabrys B., Musial K. Foundations and modeling of dynamic networks using dynamic graph neural networks: A survey // IEEE Access. 2021. Vol. 9. P. 79143–79168.
3. Bergantios G., Vidal-Puga J. J. The optimistic TU game in minimum cost spanning tree problems // International Journal of Game Theory. 2007. Vol. 36. No 2. P. 223–239.
4. Yu L., Xie W., Xie D., et al. Deep reinforcement learning for smart home energy management // IEEE Internet of Things Journal. 2019. Vol. 7. No 4. P. 2751–2762.
5. Watkins C. J. C. H., Dayan, P. Q-learning // Machine learning. 1992. Vol. 8. No 2. P. 279–292.

UDC 510

Xu Y.

Cooperative network game with partner sets

Recommended for publication by professor Petrosyan L. A.

1. Introduction. Cooperative games on networks are an important branch of game theory. Recent years have seen in-depth network game studies. Paper [1] proposed a special cooperative mechanism ensuring time consistency of the characteristic function. In [2], partner sets in network games are investigated. Based on these studies, this paper innovatively proposes a payoff-allocation method according to the network structure in a cooperative differential game model. Specifically, players select a partner sets, partner sets cooperate for maximum total payoff, divide it by Shapley value and individuals get share of payoffs according to the structure in each partner set.

2. Model. Here we describe the network with partner sets and the introduce a multi-stage process for the game.

2.1. Game on network with partner sets. Consider a class of n-player differential games on network with time horizon $[t_0, T]$. Denote the set of players by $N = \{1, \ldots, n\}$, the set of nodes in network also by N, for each node represents a player. Arcs in the undirected network show communication between players. The set of arcs is denoted by L, where $L = \{(i,j) \mid i,j \in N,\ i \text{ and } j \text{ can communicate with each other},\ i \neq j\}$. Denote the set of nodes that are directly connected with node i by $\widetilde{K}(i)$, $\widetilde{K}(i) = \{j \in N | (i,j) \in L\}$.

Consider another class of subsets of N, namely the partner sets selected by each player. Suppose each player i chooses one partner set M_i at the beginning of the game. Note that i must belong to M_i and act as the leader. For any $j \in M_i$, $j \neq i$, there exists a path connecting i and j in M_i, $i = 1, \ldots, n$. According to network G, we can construct the set of all possible partner sets of player i, $i = 1, \ldots, n$. Denote it as $\widetilde{\mathcal{M}}_i = \{M_i | M_i \subseteq N, \{i\} \in M_i,$ for any $j \in M_i,$ there exists at least one path in M_i from i to $j\}$. There are n partner sets in total, $M_1, \ldots, M_n$, and their intersections can be non-empty.

Xu Yijia – graduate student, Saint Petersburg State University; e-mail: st122181@student.spbu.ru, phone: +7(963)249-46-25

The control vector of player i is denoted as $u_i(t)$, $u_i(t) = (u_{ik}(t))_{k\in N}$, where $u_{ik}(t)$ represents the control function of player i in partner set M_k. For M_k that i does not belong to, $u_{ik}(t) = 0$; otherwise, $u_{ik}(t) > 0$. Suppose there is a restriction that total control of a player is limited to 1, that is $\sum_{k=1}^{n} u_{ik}(t) \leq 1$, $i \in N$, $\forall t \in [t_0, T]$. Denote the set of all possible control vectors of player i by U_i, $U_i = \{u_i(t) \mid u_i(t) = (u_{ik}(t))_{k\in N}, u_{ik}(t) \geq 0, k \in N, t \in [t_0, T], \sum_{k=1}^{n} u_{ik}(t) \leq 1, t \in [t_0, T]\}, i \in N$.

The state variable of player i in partner set M_k is denoted as $x_{ik}(t)$, which depends on all control functions that are used in M_k. The dynamic system is in form (1) with initial conditions $x_{ik}(t_0) = x_{ik0}, i, k \in N$, $f(x_{ik}(t), u_{jk}(t), j \in N)$ is continuously differentiable in $x_{ik}(t)$ and $u_{jk}(t)$:

$$\dot{x}_{ik}(t) = f(x_{ik}(t), u_{jk}(t), j \in N). \tag{1}$$

As for payoff function H_i of player i, it is the summand share player i gets from each partner set. Payoff in this game comes from communication between two directly connected players. In section 3, we will show cooperation and imputation of partner sets, and in section 4, we will introduce the shares among players in each partner set.

2.2. Multi-stage process. Denote the game by $\Gamma = \langle N, \widetilde{\mathcal{M}}_{i,i\in N}, U_{i,i\in N}, H_{i,i\in N}\rangle$. The multi-stage game proceeds as follows: on stage one, each player i chooses his partner set M_i from $\widetilde{\mathcal{M}}_i$; on stage two, each player i gets all information of all selected partner sets $M_1, \ldots, M_n$, and for each M_k, players inside determine to be in this partner set or not. Suppose one player can be in many partner sets. After that, partner sets $M_1, \ldots, M_n$ are reconstructed and remain fixed throughout the time horizon $[t_0, T]$. Player i will stay in M_k if and only if i agrees to be in M_k and there is at least one path from player k to i where all the players on the path agree to be in M_k; otherwise, i will leave M_k. On the next stage, partner sets $M_1, \ldots, M_n$ will act as players and play a cooperative game.

3. Cooperative game of partner sets. In this section, we consider game $\overline{\Gamma}$ where each partner set acts as a player. We will define the payoff function of partner sets and the cooperative game of them. Denote the set of players by $\overline{N}$, $\overline{N} = \{M_1, \ldots, M_n\}$. In partner set M_k, denote communication payoff between i and j by $h_{ij}^k(x_{ik}(t), x_{jk}(t))$. It depends on the state variables of the two individuals and is non-negative for any $i, j \in N$, $M_k \in \overline{N}$, $t \in [t_0, T]$. The payoff of partner set M_k, denoted by $\overline{H}_k$, includes communication inside M_k and the extra payoff its individ-

uals brought from communicating in another partner set M_l. This extra payoff is added to $\overline{H}_k$ with a parameter $\alpha(M_k, M_l) \in (0,1)$. Introduce an activity efficiency factor δ as a discount rate when one more participant joins the partner set. Denote d_{ik} as "the shortest distance between individual i and leader k", $\alpha_{ik} = \alpha^{d_{ik}}$, where α is transfer efficiency, represents the impact of transferring along the shortest path. The payoff function $\overline{H}_k(x_0, u) = \sum_{i \in M_k} \alpha_{ik} \Bigg(\sum_{j \in M_k \cap \widetilde{K}(i)} \delta^{|M_k|-1} \int_{t_0}^{T} h_{ij}^k(x_{ik}, x_{jk}) dt + \sum_{M_l \in \overline{N} \setminus \{M_k\}, i \in M_l} \alpha(M_k, M_l) \sum_{s \in M_l \cap \widetilde{K}(i)} \delta^{|M_l|-1} \int_{t_0}^{T} h_{is}^l(x_{il}, x_{sl}) dt \Bigg)$.

3.1. Cooperation. Now consider a cooperative game in characteristic function form. To achieve group optimality, players (partner sets) maximize their joint payoff $\sum_{M_j \in \overline{N}} \overline{H}_j$ respect to $u_1(t), i \in N$ and subject to dynamics (1). Suppose the optimal strategies $u_i^*(t)$ lead to the optimal trajectory $x_{ik}^*(t)$, $i \in N$, $M_k \in \overline{N}$, $t \in [t_0, T]$. Define the characteristic function to measure the value when all players act as one coalition:

$$v(\overline{N}) = \sum_{M_j \in \overline{N}} \Bigg[\sum_{i \in M_j} \Bigg(\sum_{l \in M_j \cap \widetilde{K}(i)} \delta^{|M_j|-1} \int_{t_0}^{T} h_{il}^j(x_{ij}^*, x_{lj}^*) dt + \sum_{\substack{M_k \in \overline{N} \setminus \{M_j\}, \\ i \in M_k}} \alpha_{(M_j, M_k)} \sum_{q \in M_k \cap \widetilde{K}(i)} \delta^{|M_k|-1} \int_{t_0}^{T} h_{iq}^k(x_{ik}^*, x_{qk}^*) dt \Bigg) \Bigg], \tag{2}$$

for any $S \subseteq \overline{N}$, here use the same special settings as paper [1], maintain the cooperative strategies for all players along the cooperative trajectory when calculating the values of characteristic function for coalitions.

$$v(S) = \sum_{M_k \in S} \Bigg[\sum_{i \in M_k} \alpha_{ik} \Bigg(\sum_{j \in M_k \cap \widetilde{K}(i)} \delta^{|M_k|-1} \int_{t_0}^{T} h_{ij}^k(x_{ik}^*, x_{jk}^*) dt + \sum_{\substack{M_l \in S \setminus \{M_k\}, \\ i \in M_l}} \alpha(M_k, M_l) \sum_{s \in M_l \cap \widetilde{K}(i)} \delta^{|M_l|-1} \int_{t_0}^{T} h_{is}^l(x_{il}^*, x_{sl}^*) dt + \alpha(S) \sum_{\substack{M_l \in \overline{N} \setminus S, \\ i \in M_l}} \alpha(M_k, M_l) \sum_{s \in M_l \cap \widetilde{K}(i)} \delta^{|M_l|-1} \int_{t_0}^{T} h_{is}^l(x_{il}^*, x_{sl}^*) dt \Bigg) \Bigg], \tag{3}$$

where $\alpha(S) \in (0,1)$, $\alpha(S)$ increases as S becomes larger. The characteristic function is superadditive and time consistent.

3.2. Shapley value imputation. To divide the joint maximum payoff, we consider imputation distribution based on Shapley values [3]. The Shapley value for player (Partner set) $M_k \in \overline{N}$ is given by: $\mathrm{Sh}_k(x_0, T-t_0) = \sum_{S\subset\overline{N}, S\ni M_k} \frac{(|S|-1)!(n-|S|)!}{n!} \times [v(S) - v(S\setminus\{M_k\})]$.

4. Payoff division inside partner sets. In this section, we allocate the Shapley value of each partner set to its members based on the level structure. Denote the share individual i gets from M_k by Sh_{ki}, it includes two parts.

4.1. Bonus of individual. The first part is the value of the characteristic function when individual i plays alone. Use the same parameters as in section 3 and define a cooperative game in partner set M_k. $v_k(\{i\})$ as follows is considered to be the bonus of i got from M_k:

$$v_k(\{i\}) = \alpha_{ik} \cdot \alpha(\{i\}) \cdot \Bigg(\sum_{r\in M_k\cap\widetilde{K}(i)} \delta^{|M_k|-1} \int_{t_0}^{T} h_{ir}^{k}(x_{ik}^{*}, x_{rk}^{*}) + \sum_{\substack{M_l\in\overline{N}\setminus\{M_k\},\\ i\in M_l}} \alpha_{(M_k,M_l)} \sum_{s\in M_l\cap\widetilde{K}(i)} \delta^{|M_l|-1} \int_{t_0}^{T} h_{is}^{l}(x_{il}^{*}, x_{sl}^{*})dt \Bigg). \tag{4}$$

4.2. Level structure. The second part is structural gratuity. Assign levels to players in partner set M_k according to network G. As set up in section 2, in M_k, k is the leader (level 1). Players directly connected to k are level 2, and the remaining individuals directly connected to level 2, are level 3. This process continues until all members of M_k are ranked. Construct a directed network G_k based on G and the level structure of M_k. Take the subnet of G restricted to M_k, reserve the arcs connecting individuals at different levels, and direct them from the smaller level to the larger level individuals. Between M_k and another partner set, we draw an arc from individual i in M_k to j in M_l, if i appears in M_l and i and j are directly connected in G.

4.3. Adjacency matrix. With the directed network G_k, we construct an adjacency matrix C_k, representing the connections between nodes in G_k. C_k is an $m \times m$ matrix, where $m = |M_1| + \ldots + |M_n|$. Both its rows and columns are partitioned into n blocks, corresponding to M_k, $k \in N$, respectively. Note that one individual in different partner sets are considered to be different. If there exists an arc pointing from

individual i in M_k to j in M_l, the element at the position (i_{M_k}, j_{M_l}) in C_k is set to 1; otherwise, set 0. Since C_k represents the number of paths of length 1 from a row individual to a column one, C_k^n indicates the number of paths of length n. To calculate the number of all paths of any length from a node and use it as a measure of status [4], we need to calculate matrix $P_k = \alpha C_k + \alpha^2 C_k^2 + \ldots$, where α refers to the transfer efficiency in network G, and look for its row sum. Due to the property of network G_k, the maximum length of paths from one node does not exceed n, so we only need to calculate up to the n-th power term $\alpha^n C_k^n$.

4.4. Status share. Regarding the row-sum vector of matrix P_k as each individual's status in M_k, we denote it as $\boldsymbol{\gamma}_k$ and individual i's status as γ_{ki}. The status proportion of individual i in M_k is given by $\frac{\gamma_{ki}}{\sum_{j \in M_k} \gamma_{kj}}$. This proportion is used to allocate the second part of the Shapley value in partner set M_k, ensuring that individuals with higher status receive a larger share.

The share Sh_{ki} that individual i obtains from M_k is then equal to $v_k(\{i\}) + \frac{\gamma_{ki}}{\sum_{j \in M_k} \gamma_{kj}} (\mathrm{Sh}_k - \sum_{j \in M_k} v_k(\{j\}))$, where $v_k(\{i\})$ represents the value that individual i can generate independently in the context related to M_k, and Sh_k is the total value available within M_k.

5. Result. Subsequently, player i's total payoff in the game equals the sum of shares from each partner set, i.e., $H_i = \sum_{M_k \in \overline{N}, M_k \ni i} \mathrm{Sh}_{ki}$. It is easy to see that $\sum_{i \in N} H_i = v(N)$. This approach provides a refined model for analyzing the distribution considering the influence of a players status in partner sets on their payoff in the overall game.

References

1. Bulgakova M. A., Petrosyan L. A. About one multistage non-antagonistic network game // Vestnik SPbU. Applied Mathematics. Computer Science. Control Processes. 2019. Vol. 15. No 4. P. 603–615.
2. Petrosyan L. A., Yeung D., Pankratova Y. B. Cooperative differential games with partner sets on networks // Trudy Inst. Mat. i Mekh. UrO RAN. 2021. V. 27. No 3. P. 286–295.
3. Shapley L. S. A value for n-person games // Contributions to the Theory of Games II / Kuhn H., Tucker A. eds. Princeton: Princeton University Press. 1953. P. 307–317.
4. Katz L. A new status index derived from sociometric analysis // Psychometrika. 1953. Vol. 18. P. 39–43.

UDC 519.83

Yu X.

An evolutionary games model with time delays

Recommended for publication by associate professor Gubar E. A.

1. Introduction. During the epidemic, the spread of incorrect information will affect public health measures and social stability. In the process of information dissemination, the population divides into two subgroups: those who read incorrect information and correct information. Let x_f and x_r represent the proportion believing incorrect and correct information. The proportion of these two groups changes over time, and $x_f(t) + x_r(t) = 1$, $\forall t$. Agents who follow the correct information tend to take appropriate precautions, while agents who follow incorrect information may not take any protective measures. When any two agents interact, they exchange information and make decisions: agents interacting with those using different strategies may shift their own, while those meeting like-minded individuals maintain their original choices.

This interaction process is analyzable through evolutionary game theory, where agent decisions depend on both information and system state. In turn, these decisions will change the state of the system, forming a feedback loop. In this game, participants are agents from both subgroups, with strategy set $S = \{f, r\}$, representing belief in incorrect (f) or correct (r) information.

The decisions and proportion changes of these subgroups will affect the payoff matrix, which is closely related to the infection status in the population. Gubar et al. [1] proposed a payoff matrix $A(I)$:

$$A(I) = I \begin{pmatrix} c_1 & k_f \\ k_r & c_2 \end{pmatrix} + (1 - I) \begin{pmatrix} \overline{c}_1 & \overline{k}_f \\ \overline{k}_r & \overline{c}_2 \end{pmatrix}, \tag{1}$$

where c_i $(i = 1, 2)$ represents treatment costs for those following incorrect and correct information when infected, while $\overline{c}_i$ represents prevention costs when not infected, with $c_i \gg \overline{c}_i$; k_f and k_r denote treatment costs for followers of incorrect and correct information, while $\overline{k}_f$ and $\overline{k}_r$ represent prevention costs for uninfected individuals.

Yu Xiao – graduate student, Saint Petersburg State University; e-mail: st115845@student.spbu.ru, phone: +7(812)363-65-37

In epidemiological modeling, the SIR framework is a foundational model for disease transmission. The population is divided into susceptible (S), infected (I), and recovered (R) compartments with positive initial values. The dynamics governed by the following system of differential equations:

$$\dot{S} = -\beta SI, \qquad \dot{I} = \beta SI - \sigma I, \qquad \dot{R} = \sigma I. \tag{2}$$

Where β is the infection rate, representing the probability of a susceptible individual becoming infectious after contact, and σ is the recovery rate, indicating the rate at which infectious individuals recover.

2. Model framework. In the framework of evolutionary game theory, decision delays occur when agents make strategy choices. Specifically, the agent needs time to weigh the costs and benefits of choosing correct and incorrect information, and then make a strategy choice. This decision-making process is not completed instantly and requires a certain reaction time, so information dissemination influences behavioral responses with time delays. It is very important to discuss how this delay affects the dynamic evolution of the system. Therefore, based on system (2), we focus on the time delay effect, and explore the evolutionary stability under different delay conditions through numerical simulation.

We assume that when an agent initiates a game (i.e., exchanges information with other agents and begins the strategy selection process) at time t, the actual strategy execution and subsequent payoff realization occur after a delay of τ time units. Correspondingly, the expected payoff of the strategy is determined at that delayed moment. Therefore, the expected payoff of the current strategy depends on both the current and previous strategy distributions.

To model this process, we use a replicator dynamics equation with time delay. The expected benefits of believing incorrect and correct information at time t are $U_f(t,\tau)$ and $U_r(t,\tau)$ respectively, with average benefit $\bar{U}(t,\tau)$ and delay τ as shown below:

$$U_f(t,\tau) = ax_f(t-\tau) + b(1-x_f(t-\tau)), \tag{3a}$$
$$U_r(t,\tau) = cx_f(t-\tau) + d(1-x_f(t-\tau)), \tag{3b}$$
$$\bar{U}(t,\tau) = x_f(t)U_f(t-\tau) + (1-x_f(t))U_r(t-\tau), \tag{3c}$$

where the parameters a, b, c, and d are derived from the payoff matrix

$A(I)$ as shown in equation (1). Specifically: $a = Ic_1 + (1-I)\overline{c}_1$, $b = Ik_f + (1-I)\overline{k}_f$, $c = Ik_r + (1-I)\overline{k}_r$, $d = Ic_2 + (1-I)\overline{c}_2$.

According to Miekisz's study [2], assuming ϵ proportions of people play games within a very short time ϵ, we formulate an equation for the frequency of strategy f:

$$x_f(t+\epsilon) - x_f(t) = \epsilon \frac{x_f(t)[U_f(t-\tau) - \bar{U}(t-\tau)]}{1-\epsilon+\epsilon\bar{U}(t-\tau)}.$$

Divide both sides of the equality by ϵ and apply the limit $\epsilon \to 0$,

$$\begin{aligned}
\lim_{\epsilon\to 0} \frac{x_f(t+\epsilon) - x_f(t)}{\epsilon} &= \lim_{\epsilon\to 0} \frac{x_f(t)[U_f(t-\tau) - \bar{U}(t-\tau)]}{1-\epsilon+\epsilon\bar{U}(t-\tau)} \\
\Rightarrow \frac{dx_f(t)}{dt} &= x_f(t)[U_f(t-\tau) - \bar{U}(t-\tau)] \\
&= x_f(t)(1-x_f(t))[U_f(t-\tau) - U_r(t-\tau)] \\
&= -(c-a+b-d)x_f(t)[1-x_f(t)][x_f(t-\tau) - x_f^*].
\end{aligned}$$

In addition, since the delayed effect of information propagation will change the infection rate, and then change the proportion of susceptible (S), infected (I), and recovered (R) individuals, we introduce a new infection rate $\beta_f = \beta(1+vx_f)$, where $v \geq 0$ represents the impact of incorrect information on the infection rate, so that a feedback mechanism can be established between the replicator dynamics and the SIR model:

$$\dot{S} = -\beta_f SI, \qquad \dot{I} = \beta_f SI - \sigma I, \qquad \dot{R} = \sigma I. \tag{4}$$

By combining the time-delayed replication dynamics with the classic SIR model, we establish the following dynamic system:

$$\begin{cases}
\dot{x_f}(t) = -(c-a+b-d)x_f(t)[1-x_f(t)][x_f(t-\tau) - x_f^*], \\
\dot{S} = -\beta_f SI, \\
\dot{I} = \beta_f SI - \sigma I, \\
\dot{R} = \sigma I,
\end{cases} \tag{5}$$

where $c-a+b-d = (k_r + k_f - c_1 - c_2)I + (\overline{k}_r + \overline{k}_f - \overline{c}_1 - \overline{c}_2)(1-I)$ is derived from the payoff matrix $A(I)$ as shown in equation (1).

3. Numerical simulations of feedback-coupled SIR transmission model. To support the theoretical findings, we conduct numerical

calculations with the following parameters: $\beta = 0.35$, $\sigma = 0.2$, and initial states $S(0) = 0.9$, $I(0) = 0.1$, $R(0) = 0$, $x_f(0) = 0.7$, $x_r(0) = 0.3$. The simulation runs over time interval [0, 90] with payoff matrix parameters $c_1 = 10$, $c_2 = 8$, $\overline{c}_1 = 2$, $\overline{c}_2 = 1.5$, $k_f = 5$, $k_r = 3$, $\overline{k}_f = 1$, and $\overline{k}_r = 0.5$.

For system stability analysis, we examined the long-term evolutionary characteristics of incorrect and correct information followers at three key time delays: $\tau = (0, 0.8, 3)$. The results and analysis follow below.

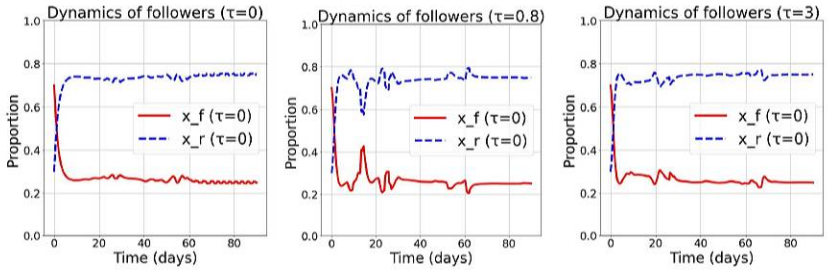


Figure 1. Replicator dynamics with different time delays $\tau = 0, 0.8, 3$

In this propagation process, time delay significantly affects the system dynamics, but does not affect the final steady state. When there is no time delay, the system quickly stabilizes. When there is a moderate delay, oscillations occur, but the system eventually reaches stability. When the delay is greater, periodic fluctuations emerge, yet the long-term trend remains unchanged: the influence of incorrect information is a minority, and correct information dominates.

We also analyzed the evolutionary trends of susceptible (S), infected (I), and recovered (R) populations, identifying the peak infection level (maximum value of I) and its corresponding time.

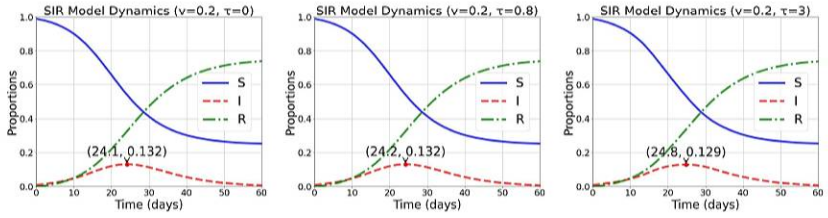


Figure 2. SIR model dynamics with time delays $\tau = (0, 0.8, 3)$ with v=0.2

When the time delay increases to low delay ($\tau = 0.8$), the peak of the

infection curve rises slightly, and the time to reach the peak is delayed; when the time delay increases to high delay($\tau = 3$), the infection peak decreases and the time to reach the peak is shortened. This shows that high delay enhances the feedback mechanism of the system, and the impact of the past state on the current state becomes more significant, thereby suppressing the growth rate of the number of infected people.

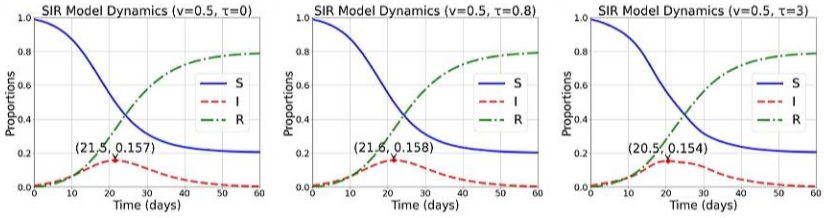


Figure 3. SIR Model Dynamics with time delays $\tau = (0, 0.8, 3)$ with v=0.5

When misinformation impact coefficient v rises from 0.2 to 0.5, infection peaks appear earlier with slightly higher values. This shows that stronger misinformation accelerates disease spread, causing worse outbreaks. Additionally, susceptible groups(S) decline more rapidly while recovered populations(R) increase faster, ultimately reaching higher levels. In summary, elevated v values triggering quicker, more severe epidemics.

References

1. Gubar E., Taynitskiy V., Dahmouni I. The impact of fake news on infection dynamics in Pandemic control: An evolutionary SIR model // IFAC-PapersOnLine. 2023. Vol. 56. No 2. P. 1778–1783.

2. Miekisz J. Evolutionary game theory and population dynamics // Multiscale Problems in the Life Sciences: From Microscopic to Macroscopic. Springer Lecture Notes in Mathematics. 2008. Vol. 1940. P. 269–316.

UDC 519.83

Zheng Z.

Impact of interaction time on overprotective behavior during epidemics: an evolutionary model

Recommended for publication by associate professor Gubar E. A.

1. Introduction. Most classic evolutionary games assume an infinite population where individuals play pairwise games. Interactions between different pairs often take different amounts of time, this will affect the game. During the epidemics, people frequently discuss the epidemic on social media, exchanging their perspectives. The longer they interact, the more they can learn from the strengths of others. They adjust their behavior through the information they exchange, which can influence the course of the epidemic. In our model, the classic SIR model [1] is incorporated to describe the epidemic dynamics. Following [2], a game is constructed between two different perspectives within the epidemic context. Then we develop a new evolutionary dynamic with interaction time based on the replicator dynamic [3]. The results show that this dynamic leads to a different evolutionary outcome. As interaction time increases, people tend to choose overprotective behaviors when the epidemic situation is serious, and tend to choose reckless behaviors when the situation has eased.

2. Model Description. During an epidemic, people adopt different behaviors: some overprotect and isolate, while others act recklessly without protective measures. The distribution of human behaviors influences the infection rate, thereby affecting the epidemic process. The more reckless individuals there are, the higher the infection rate in the population. However, the proportion of infected individuals in the epidemic also impacts human behavior: the higher the number of infections, the more people tend to adopt protective behaviors.

In our model, we use the classic Susceptible–Infected–Recovered model [1] to describe the epidemic environment:

Zheng Zishen – graduate student, Saint Petersburg State University; e-mail: 122465@student.spbu.ru, phone: +7(812)363-65-37

$$\begin{aligned} \dot{S} &= -\beta SI, \\ \dot{I} &= \beta SI - \sigma I, \\ \dot{R} &= \sigma I, \end{aligned} \tag{1}$$

where $S(t)$, $I(t)$, $R(t)$ indicate the proportion of susceptible, infected, and recovered people, respectively. Initial states are given by $S(0) = S^0 > 0$, $I(0) = I^0 > 0$ and $R(0) = R^0$. Value β and σ indicate infection and recovery rates ($0 < \beta < 1$, $0 < \sigma < 1$).

Following [2], we construct a evolutionary game with N agents, where N is large, but finite. Each agent has the same strategy space S, which consists of two behaviors: *overprotection* and *recklessness*. The former indicates that individuals take all protective measures against the epidemic, such as staying indoors and avoiding social activities. The latter, on the contrary, represents the complete absence of protective measures. Then we define the following payoff matrix $A(I)$, which depends on the Infected subgroup:

$$A(I) = I \begin{pmatrix} c_1 & k_1 \\ k_2 & c_2 \end{pmatrix} + (1 - I) \begin{pmatrix} \bar{c}_1 & \bar{k}_1 \\ \bar{k}_2 & \bar{c}_2 \end{pmatrix}, \tag{2}$$

where c_1 and c_2 are the costs of treatment ($c_1 < c_2$) in the case where the followers of overprotective and reckless behaviors are in the infected state. $\bar{c}_1$, $\bar{c}_2$ are the cost of preventive measures ($\bar{c}_1 > \bar{c}_2$) in the case where the agents are not in the infected state, ($c_i << \bar{c}_i$), with $i = 1, 2$. When two individuals of the same type interact, nothing happens, and they continue to do what they were doing. When the two different types of individuals interact, there is a probability that they may change their options and adopt the behavior of their opponents. While the probability of switching from *overprotection* to *recklessness* is ν. The probability of going from *recklessness* to *overprotection* is μ. And

$$\begin{aligned} k_2 &= c_1\mu + c_2(1 - \mu), \;\; k_1 = c_2\nu + c_1(1 - \nu), \\ \bar{k}_2 &= \bar{c}_1\mu + \bar{c}_2(1 - \mu), \;\; \bar{k}_1 = \bar{c}_2\nu + \bar{c}_1(1 - \nu). \end{aligned} \tag{3}$$

Following [4], we assume that all interactions take some time and interactions between different pairs take different amounts of time. Thus we construct an interaction time matrix T:

$$T = \begin{pmatrix} \tau_1 & \tau_2 \\ \tau_2 & \tau_1 \end{pmatrix}, \tag{4}$$

where τ_1 and τ_2 are the interaction times when individuals of the same type and different types meet, respectively ($\tau_1 < \tau_2$).

3. Evolutionary dynamic with interaction time. According to Sandholm's definition[3], the procedures agents follow in deciding which strategies to switch to are called revision protocols. In order to investigate the effect of interaction time on the evolutionary dynamics of our model. We added the interaction time as a new input and defined the new revision protocol.

Definition 1. A revision protocol $\tilde{\rho}$ is a map $\tilde{\rho} : \mathbf{R}^n \times \mathbf{R}^n_+ \times X \to \mathbf{R}^{n\times n}_+$. The scalar $\tilde{\rho}_{ij}(\pi, \tau, x)$ is called the conditional switch index from strategy $i \in S$ to strategy $j \in S$ given payoff vector π, interaction time vector τ and population state x.

Each $x \in \{(x_1, \ldots, x_n) \mid x_i \geq 0, \sum x_i = 1\}$ represents a population state in which the proportion of agents who use the strategy i is indicated as x_i. The payoff vector π and the interaction time vector τ are defined similarly, with π_i and τ_i representing the payoff and interaction time of the agent using strategy i, respectively.

We suppose that an agent who receives a revision opportunity chooses an opponent at random and observes her strategy. If the opponent's payoff and interaction time are higher than his own, he will choose to imitate his opponent's strategy. We also assume that each agent's payoffs and interaction time only depend on opponents behavior through the distribution of opponents choices. Thus we can construct this protocol:

$$\tilde{\rho}_{ij}(\pi, \tau, x) = x_j\Big([\pi_j(x) - \pi_i(x)]_+ + [r(\tau_j(x) - \tau_i(x))]_+\Big). \tag{5}$$

If our agent is playing strategy i and the opponent strategy j, the agent switches from i to j with index proportional to the difference between the payoff and the interaction time impact of strategy i and strategy j. Value r denote the degree of interaction time impact, which can be interpreted as benefits per unit of interaction time.

The mean dynamic [3] generated by this revision protocol is

$$\begin{aligned}\dot{x_i} &= \sum_{j\in S} x_j\tilde{\rho}_{ji}(\pi,\tau,x) - x_i\sum_{j\in S}\tilde{\rho}_{ij}(\pi,\tau,x)\\ &= x_i\sum_{j\in S} x_j[(\pi_i(x)-\pi_j(x))+r(\tau_i(x)-\tau_j(x))]\\ &= x_i[(\pi_i(x)-\bar{\pi}(x))+r(\tau_i(x)-\bar{\tau}(x))].\end{aligned} \tag{6}$$

The first term of this expression is the standard form of the replicator dynamic. The latter term we can interpret as a time impact.

Considering our model, we can write the payoff and interaction time of strategy *overprotection*:

$$\pi_1(x) = e_1A(I)x,\ \tau_1(x) = e_1Tx, \tag{7}$$

where $e_1 = (1,0)$. As interaction time increases, paired agents may acquire new and useful information, learning from each other's strengths. This provides them with positive benefits. The higher the number of infected individuals, the more attention people pay to the epidemics, and the greater the impact of interaction time becomes. Therefore, r is positive and depends on the proportion of infected people I, and we set $r(I) = Ir_1 + (1-I)r_2$, $(r_1 > r_2 > 0)$.

Then we substitute equation (7) into equation (6) yields the evolutionary dynamic of overprotective behavior:

$$\dot{x_1} = x_1(e_1 - x)\big(A(I) + r(I)T\big)x. \tag{8}$$

Thus, we have an epidemic dynamic, coupled with an evolutionary dynamic, which defines the distribution between followers of overprotective and reckless behaviors.The associated extended dynamics are defined as follows:

$$\begin{aligned}\dot{S}(t) &= -\beta SI,\\ \dot{I}(t) &= \beta SI - \sigma I,\\ \dot{R}(t) &= \sigma I,\\ \dot{x_1}(t) &= x_1(e_1 - x)\big(A(I) + r(I)T\big)x.\end{aligned} \tag{9}$$

The higher the proportion of overprotected people, the lower the infection rate in the population. Thus we set infection rate in the population as $\beta = x_1\beta_1 + x_2\beta_2$, $(\beta_2 > \beta_1 > 0)$.

4. Numerical simulation and analysis of the results. In order to study the impact of interaction time on the evolution of both behaviors and the overall epidemic process, we simulated the system (9).Then we set $c_1 = -7$, $c_2 = -10$, $\bar{c}_1 = -0.1$, $\bar{c}_2 = -0.05$, $\beta_1 = 0.1$, $\beta_2 = 0.3$, $u = 0.3$, $v = 0.3$, $r_1 = 0.1$ and $r_2 = 0.005$. The figures show us that both the SIR model and the dynamics of behaviors are affected with increasing τ_{12}. As τ_{12} increases, the peak in the proportion of infected people becomes higher, and even at the peak of infection there are still people who choose reckless behavior, and as the number of infected people decreases in the later stages, the proportion of people who choose reckless behavior becomes higher.

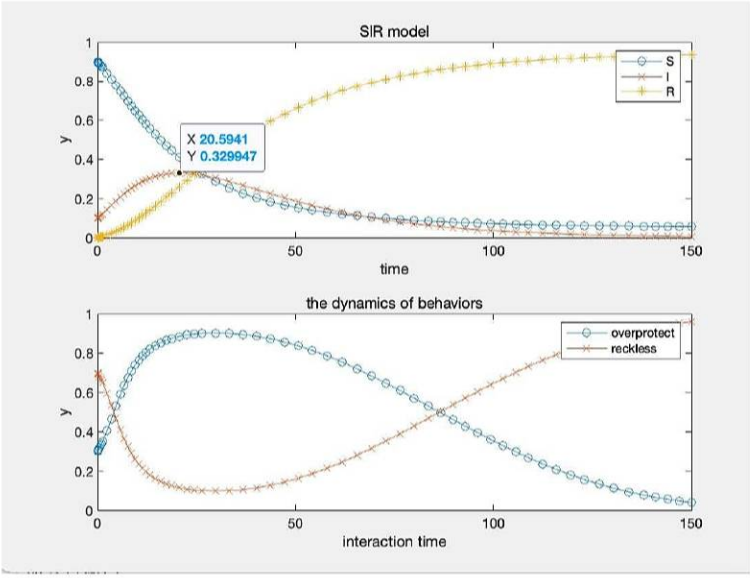


Figure 1. SIR model and evolution of behavior when $\tau_1 = \tau_2 = 1$

The natural interpretation of this phenomenon is easy to understand; reckless people don't restrain themselves, they learn the medical treatment of overprotective people after infection, and the more they can learn as they communicate more. This makes more irresponsible people follow this behavior. And at the end of the epidemic, people naturally like freedom and do not want to be protective anymore.

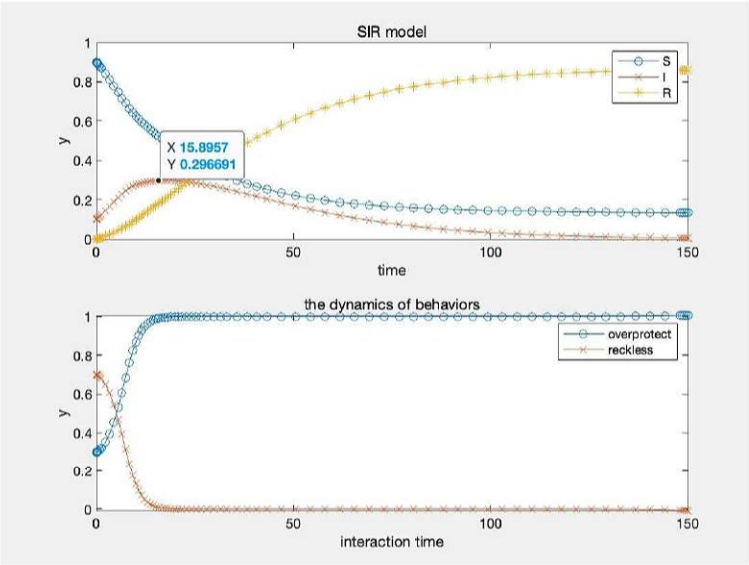


Figure 2. SIR model and evolution of behavior when $\tau_2 = 20$

References

1. Kermack W. O., McKendrick A. G. A contribution to the mathematical theory of epidemics // Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character. 1927. Vol. 115. No 772. P. 700–721.

2. Gubar E., Taynitskiy V., Dahmouni I. The impact of fake news on infection dynamics in Pandemic control: An evolutionary SIR model // IFAC-PapersOnLine. 2023. Vol. 56. No 2. P. 1778–1783.

3. Sandholm W. H. Population Games and Evolutionary Dynamics. Cambridge, MA: MIT Press, 2010. 616 p.

4. Křivan V., Cressman R. Interaction times change evolutionary outcomes: Two-player matrix games // Journal of Theoretical Biology. 2017. Vol. 416. P. 199–207.

Contents

Plenary session **11**

Invited speakers 11

Sidorin A. O. Control processes in fundamental science 11

Abbasov M. E. Searching for optimal trajectory on the terrain 13

Vashhenkov K. V., Ponomarev Yu. Yu. The platform of the XSQUARE-LCDP company is a Russian PostgreSQL-oriented platform for low code systems – the path to the future for novice developers. Brief introduction 27

1. Mathematical theory of control processes **37**

Evtina D. S., Zhabko A. P. Estimation of solutions of neutral type equations with incommensurate delays 37

Zhivulin E. A., Grishkin V. M. Construction of a vessel's route using minimization methods for the purpose of economical fuel consumption 42

Kamintseva V. S. Construction of an optimal regulator for systems with disturbances 47

Kosterov M. A. Computation of parabolic approximation-estimation tests' coefficients using QR-decomposition 55

Semenova A. Yu. Multi-criteria optimization of control parameters for the dynamics of a charged particle beam in a linear accelerator 62

Sokolov V. M. Lyapunov – Krasovsky functional with a prescribed derivative for the wave equation 71

Estrada Jimenez O. Algorithm to determine the sequence of movements necessary to change the position of stacked containers 78

2. Mathematical methods in mechanics and physics **85**

Vavrenyuk V. V. A fuzzy controller in the problem of solar sail stabilization 85

Viktorov B. V. Algorithm for a group of robots visiting a set of points 95

Vyaltseva V. V. Estimation of induced anisotropy of effective properties for materials with cracks 100

Gumerov I. I., Kholodova S. E. Development of methods for mathematical modeling of dynamic processes in a rotating electrically conductive liquid medium 105

Kikenov A. A. Modeling of controlled motion of a tiltrotor tricopter 112

Kniazev N. A., Levinskij D. V. Changes accounting in magnetic fields of dipoles at modeling transverse particle motion in a synchrotron 118

Mullayanov T. O. The influence of the shape of the nanoscale relief on the stress distribution near the free surface of a solid ... 124

Ruch'ev I. A., Shimanchuk D. V. Machine learning methods for solving the inverse problem of robotic manipulator kinematics with excessive numbers of freedom degrees 130

Suleymanov V. N. Simulation of the shape of a solar sail under the influence of Sun radiation pressure in the vicinity of the libration point L1 139

Yakimova I. O., Shuvalov G. M. Modeling of crack propagation under stress and corrosion 147

Li Y. Design and implementation of Multi-UAV cooperative reconnaissance and strike mission planning system 152

3. Mathematical models of medical and biological systems 157

Anikushin V. A., Redko A. Yu. Forecasting the dynamics of the COVID-19 epidemic based on percentage increases in European and Asian countries 157

Vasilev E. V. Heatmaps conjugated with the hierarchical structure of medical observations 165

Davydenko B. N. Textural analysis of micrographs of cell cultures 170

Kirichenko E. Yu., Gromov R. M., Blekanov I. S., Nazarov V. D., Denisov D. G., Lapin S. V. Methods for diagnosing anemia in laboratory tests based on machine learning 174

Kozin A. A. Classification of breast tumors by BI-RADS using neural networks 179

Krivoshein A. E., Khomenko I. E. A multi-class and two-stage approach to classification and processing of medical data 184

Morgach D. Yu. Filtering electromyographic signals to classify actions 189

Topper A. M. Development of an automated system for processing patient outpatient records 194

Shaikhislamova A. R. Mathematical modeling of mortality in the Republic of Moldova: probabilistic and statistical analysis 199

Sharasheva N. I. Predicting the prevalence of cancer 205

Kamguia F. H. AI-explainable medical image comparison 211

Wang B., Hu R., Xia Y. Rumor spread model and impact assessment in public health mergencies 218

4. Information and computer technologies 223

Avramenko P. A. Approximate graph coloring in telecom networks 223

Arzha A. N. Application of autoencoders to detect anomalous time series in smart meter data 229

Aristakesyan T. A., Blekanov I. S. Methods for decomposition of user discussion topics based on dynamic topic modeling 236

Babkina A. V., Vakhitov B. D., Devrishev T. E., Mishin I. V., Petrenko A. A. Tuberculosis signs detection on chest x-ray images 243

Babkina A. V., Konev D. V. Kidney segmentation on ultrasound images 251

Bagramov N. I. Automatic generation of scientific paper annotations based on natural language processing 259

Vasilev V. A. Quantization of weights in key components of transformer architectures 265

Vashukova A. M. Application of neural ordinary differential equations to modeling dynamic systems 270

Gavrilov N. O., Korkhov V. V., Pen E. A., Tokarev A. E. Algorithm for constructing a knowledge graph from patent data..... 275

Golofastov L. D., Mironov A. V. Implementation and power analysis of goodness-of-fit criteria for exponential distribution ... 282

Goloshhapova A. M. Application of hyperparameters optimization methods to improve the accuracy of ad clicks prediction models 290

Zharikov D. S., Blekanov I. S. Methods for distributed download of medical data in DICOM format 295

Zubrickii I. V. An area-efficient method for implementing hardware support for cryptographic information protection 301

Ibatullin E. V., Lamzin N. A. Data Mesh architecture for big data environments 309

Izmailov N. O. Training a binary neural network using its continuous counterpart 313

Kruglov N. A. A software package for solving transportation optimization problems 319

Kuznecova D. S. Study of particle swarm optimization (PSO) initialization methods for solving global optimization problems..... 325

Kuklo R. A. Research of data collecting, processing and compression techniques using edge computing in AIoT..... 333

Lavrukhin M. Yu. Application of a genetic algorithm for optimizing three-dimensional trajectories over terrain 340

Mironov A. V. Implementation and power analysis of goodness-of-fit tests with the Weibull distribution..... 346

Moshkin L. S. Using approximation and evaluation criteria to improve the HDBSCAN algorithm 352

Muradyan D. S., Alimov P. G. Implementation and integration of Gated Recurrent Unit architecture in the neural network approximation package for differential equations DEGANN..... 360

Petrova A. M. Methods for recognizing different types of food products in images 366

Ponomarev Yu. Yu. To the question of accurate estimates of tree characteristics in modeling ultra-large graphs of technological and natural networks with specified statistics of vertices, edges and signal transmission 372

Popov A. P. Methods for creating LLM-based intelligent assistants for analyzing open medical data 379

Pokhabov I. I., Gorikhovskii V. I. Graph based goodness-of-fit tests 385

Saperov D. P. Assessing the impact of world events on the cryptocurrency market 391

Smirnov D. O., Blekanov I. S. Techniques for large language model adaptation in the LegalTech domain 397

Snapkov M. Yu., Shmakov A. A. Application of social network data in financial instruments analysis tasks 405

Solovjev D. N. Home automation system using a decentralized approach in data storage and processing 411

Chaikin G. A. Synthetic user data generation from Q&A data ... 417

Shabalov N. A., Boiko A. V., Malkov A. V. Machine learning algorithms for the pricing model in CMTPL 422

Shaidurov V. D., Korkhov V. V., Polovkov V. V., Pen E. A., Karel'skij K. V. Development of a system for analyzing and mapping solid minerals of the seabed using artificial intelligence 432

Yurevtseva P. N. Investigation of LLM sensitivity to users psychoemotional parameters based on synthetic personas profiling 441

Fan J., Liu J., Davydenko A. A. Applying MegaAgent framework for oil and gas extraction optimization 447

Fan X., Blekanov I. S., Gubar E. A. Evolutionary approach based on BERTopic and NetLogo to analyse user discussions in social media 455

Feng Y. Obtaining estimates for the rate of convergence of the gradient descent via machine learning 461

Gao T. Structural and power-law degree distribution analysis of public transport networks in Chinese and Russian cities based on openstreetmap data and complex network theory 467

Gao Y., Xu H. Reliability of communication network 476

Li J., Blekanov I. S. Text encoding approaches for topic modeling in the social media platform 485

Liu J., Fan J., Davydenko A. A. Swin-RepGFPN synergized YOLO: A dangerous driving behavior detection in near real-time . 491

Wang Q., Gao Z., Zhang L. Research on tomato maturity detection model based on YOLOV11 in complex lighting scenarios 499

Xu H., Petrosian O. L. Printed circuit board defect detection based on improved RT-DETR 504

Zhang L., Wu H., Petrosyan O. L. Online adaptive forecasting for web traffic with sudden drift 510

5. Control problems in social and economical systems ... 517

Wang D., Smirnov N. V., Guo Z. Model of optimal logistics management for a trading company 517

Zolotukhina P. Yu. Comparison of CPM and PERT methods for planning project deadlines using the example of organizing a photo exhibition 524

Ibraeva A. S. Performance evaluation of modified NSGA-II in the assignment task: comparison with the exact method and classical NSGA-II 528

Karabaev E. E. Analysis of poultry output using dynamic systems 536

Kremko V. V. Analysis and evaluation of the effectiveness of machine learning methods in predicting food prices 543

Mova I. A. Analysis of consumer and mortgage loans by regions of Russia: a spatial econometric approach 550

Novgorodtcev V. A. On the portfolio management problem in the case of the Bates model 557

Subbotina E. A. Optimising social media advertising costs using influence models 562

Frolova E. A. Energy-mix model for the Russian Federation 567

Cao Z. Dynamic networks with complete information and incomplete information under cross-partitioning 572

Du L. The stability of competing information flow in epidemic models 577

Kong Q., Petrosyan L. A. Dynamic network game with changing communication structure 582

Pu Y., Vostrenkova E. M. A framework integrating environmental feedback and agent opinions. Evolutionary game theory and reinforcement learning 587

Qu R., Huang R. Integrating Raft algorithm to blockchain consensus mechanism 592

Sun S., Petrosyan L. A. Dynamic hierarchical games with changing communication structure and pairwise interactions 597

Xiong X. A history-driven dynamic network game model for smart home systems: Comparative analysis with Q-learning 602

Xu Y. Cooperative network game with partner sets 606

Yu X. An evolutionary games model with time delays 611

Zheng Z. Impact of interaction time on overprotective behavior during epidemics: an evolutionary model 616

Contents **622**

Author index **629**

Именной указатель

Аббасов М. Э. 13
Авраменко П. А. 223
Алимов П. Г. 360
Аникушин В. А. 157
Аржа А. Н. 229
Аристакесян Т. А. 236
Бабкина А. В. 243, 251
Баграмов Н. И. 259
Блеканов И. С. 174, 236, 295, 397
Бойко А. В. 422
Вавренюк В. В. 85
Ван Д. 517
Васильев В. А. 265
Васильев Е. В. 165
Вахитов Б. Д. 243
Вашукова А. М. 270
Ващенков К. В. 27
Викторов Б. В. 95
Вяльцева В. В. 100
Гаврилов Н. О. 275
Го Ч. 517
Голофастов Л. Д. 282
Голощапова А. М. 290
Гориховский В. И. 385
Гришкин В. М. 42
Громов Р. М. 174
Гумеров И. И. 105
Давыденко Б. Н. 170
Девришев Т. Э. 243
Денисов Д. Г. 174
Евтина Д. С. 37
Жабко А. П. 37
Жариков Д. С. 295
Живулин Е. А. 42
Золотухина П. Ю. 524
Зубрицкий И. В. 301
Ибатуллин Е. В. 309
Ибраева А. С. 528
Измайлов Н. О. 313
Каминцева В. С. 47
Карабаев Е. Е. 536
Карельский К. В. 432
Кикенов А. А. 112
Кириченко Е. Ю. 174

Князев Н. А. 118
Козин А. А. 179
Конев Д. В. 251
Корхов В. В. 275, 432
Костеров М. А. 55
Кремко В. В. 543
Кривошеин А. Е. 184
Круглов Н. А. 319
Кузнецова Д. С. 325
Кукло Р. А. 333
Лаврухин М. Ю. 340
Ламзин Н. А. 309
Лапин С. В. 174
Левинский Д. В. 118
Малков А. В. 422
Миронов А. В. 282, 346
Мишин И. В. 243
Мова И. А. 550
Моргач Д. Ю. 189
Мошкин Л. С. 352
Муллаянов Т. О. 124
Мурадян Д. С. 360
Назаров В. Д. 174
Новгородцев В. А. 557
Пен Е. А. 275, 432
Петренко А. А. 243
Петрова А. М. 366
Половков В. В. 432
Пономарев Ю. Ю. 27, 372
Попов А. П. 379
Похабов И. И. 385
Редько А. Ю. 157
Ручьев И. А. 130
Саперов Д. П. 391
Семенова А. Ю. 62
Сидорин А. О. 11
Смирнов Д. О. 397
Смирнов Н. В. 517
Снапков М. Ю. 405
Соколов В. М. 71
Соловьев Д. Н. 411
Субботина Е. А. 562
Сулейманов В. Н. 139
Токарев А. Э. 275
Топпер А. М. 194
Фролова Е. А. 567
Холодова С. Е. 105
Хоменко И. Е. 184

Чайкин Г. А. 417
Шабалов Н. А. 422
Шайдуров В. Д. 432
Шайхисламова А. Р. 199
Шарашева Н. И. 205
Шиманчук Д. В. 130
Шмаков А. А. 405
Шувалов Г. М. 147
Эстрада Хименес О. 78
Юрьевцева П. Н. 441
Якимова И. О. 147
Blekanov I. S. 455, 485
Davydenko A. A. 447, 491
Du L. 577
Fan J. 447, 491
Fan X. 455
Feng Y. 461
Gao T. 467
Gao Y. 476
Gao Zeyu 572
Gao Zihan 499
Gubar E. A. 455
Hu R. 218
Huang R. 592
Kamguia F. H. 211
Kong Q. 582
Li J. 485
Li Y. 152
Liu J. 447, 491
Petrosyan L. A. 582, 597
Petrosyan O. L. 504, 510
Pu Y. 587
Qu R. 592
Sun S. 597
Vostrenkova E. M. 587
Wang B. 218
Wang Q. 499
Wu H. 510
Xia Y. 218
Xiong X. 602
Xu Hongfan 504
Xu Huimin 476
Xu Y. 606
Yu X. 611
Zhang Li 499
Zhang Lili 510
Zheng Z. 616

Ежегодное научное издание

ПРОЦЕССЫ УПРАВЛЕНИЯ И УСТОЙЧИВОСТЬ

Том 12(28)

Труды 56-й международной научной конференции
аспирантов и студентов

Редакторы текста на русском языке: *А. С. Ерёмин, Т. Е. Смирнова*
Редакторы текста на английском языке: *А. С. Ерёмин, М. А. Смирнова*
Компьютерная верстка: *М. Н. Смирнов, М. А. Смирнова*

Подписано в печать 24.06.2025
Усл. печ. л. 36,68. Тираж 150 экз.

198504, Россия, Санкт-Петербург, Петергоф, Университетский просп., 35